Cooperative Multipath Assisted Positioning

Vom Promotionsausschuss der
Technischen Universität Hamburg

zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

genehmigte Dissertation

von
Markus Ulmschneider

aus
Friedrichshafen

2021

Vorsitzender des Prüfungsausschusses:
Prof. Dr.-Ing. Matthias Kuhl

1. Gutachter:
Prof. Dr.-Ing. Gerhard Bauch

2. Gutachter:
Prof. Dr. Klaus Witrisal

Tag der mündlichen Prüfung:
15.12.2020

DOI: https://doi.org/10.15480/882.3299

ORCID: Markus Ulmschneider, https://orcid.org/0000-0001-7241-7057

Abstract

Location-awareness has become an important issue in modern life. In environments with good view to the sky, global navigation satellite systems can provide very accurate location estimates. Though, their positioning performance decreases significantly if the view to satellites is blocked. Indoors, no localization solution may be obtained at all.

Terrestrial signals such as from telecommunication base stations or wireless local area network routers can be used for indoor localization. However, the localization performance may suffer from multipath propagation when traditional propagation delay based algorithms are applied. The transmit signal is reflected, scattered and diffracted in the environment, distorting the signal and leading to a bias in the delay estimates.

With multipath assisted positioning, a new approach has emerged, that exploits the information in multipath components (MPCs) by treating them as line-of-sight signals from virtual transmitters. While the locations of the physical and virtual transmitters are generally unknown, they can be estimated with simultaneous localization and mapping (SLAM). One SLAM based multipath assisted positioning algorithm is Channel-SLAM, where the user location is estimated simultaneously with creating a map of physical and virtual transmitters.

The Channel-SLAM algorithm is limited to single users. There are many settings where multiple users move in the same indoor environment, such as in shopping malls or public buildings. The contribution of this thesis is to extend Channel-SLAM by cooperation among users in terms of exchanging maps of transmitter locations. This cooperation can drastically increase the positioning performance of all cooperating users.

However, Channel-SLAM is only a relative localization system, as no relation of the user location to an absolute coordinate frame is known. The transformation parameters relating the coordinate systems of a user to the coordinate system of a map the user obtains from a different user are unknown. In addition, the correspondences among transmitters observed by the user and transmitters in the map are unknown. Estimating these transformation parameters and correspondences is referred to as *map matching*.

A robust map matching scheme is crucial for such cooperation. Since the lack of diversity among virtual transmitters leads to ambiguities in map matching, the transmitter state space is augmented by information from where transmitters are visible. A transmitter is visible, if its signal can be received in a LoS condition. Two new tracking filters for Channel-SLAM are derived that map not only the locations, but also the visibility information about transmitters. The visibility information increases the robustness of Channel-SLAM by facilitating the detection of loop closures. In addition, it is used to derive a robust data association scheme. Data association in multipath assisted positioning refers to the challenge of associating signal components with transmitters, leading to robust and accurate long term SLAM. Finally, increasing the transmitter diversity with visibility information improves the positioning performance of Channel-SLAM by a more robust data association and by enabling user cooperation.

Contents

Chapter 1

Introduction

A wide range of services require precise localization. In outdoor environments with a clear view to the sky, the performance of global navigation satellite systems (GNSSs) can satisfy the need of many location-based services with respect to the localization performance [ME11]. In particular the spread of smartphones equipped with GNSS sensors has made users accustomed to knowing their current locations. In urban canyons or indoors, though, GNSS based localization suffers from multipath propagation, shadowing and blocking of signals or low received signal strengths, for example. In indoor scenarios, the performance of GNSS is poor, or no localization solution can be obtained at all.

Services based on location awareness in indoor scenarios can be found in numerous fields of life, such as in entertainment, in the health sector, in automation or in marketing. Examples for such services include the following.

- **Navigation** in airports, malls and administrative offices aims at improving the user experience by guidance on the fastest or most convenient way to a destination.
- **First-responder localization** during and after disastrous events is of crucial importance to the safety of rescue squads and affected citizens. In particular, search and rescue missions in harsh indoor areas are dependent on knowing the current location of active forces.
- **Assistive healthcare systems** can assist and relieve nurses with the location of patients. Patients with Alzheimer's disease, for example, may roam and get lost within a healthcare facility and need to be found by the staff.
- **Intelligent warehouses** allow decreasing the cost of logistics considerably by automation. Robots picking up and delivering goods in a warehouse need to know where they are and where the goods are stored.
- **Interactive and mixed reality games** are played simultaneously in a virtual and the real world. Events in the virtual world are triggered

by a player approximating a location in the physical world. For example, players can be rewarded with virtual points or posed challenges when they are at certain physical locations.

- **Augmented reality** refers to an enhancement or extension of the perception of the real world with virtual features. The placement of virtual features requires a precise localization of a user or device in the real world for a convenient user experience.

1.1 Indoor Localization

The demand for indoor localization has pushed the research in localization technologies and algorithms that do not rely on GNSSs. There are localization approaches that utilize wireless infrastructure such as radio-frequency identification (RFID) [Bd08] or ultra-wideband (UWB) [SM16] transceivers. Nevertheless, deploying and maintaining such infrastructure can be too expensive or time consuming. In contrast, radio signals from cellular base stations or wireless local area network (WLAN) routers tend to be ubiquitously available in populated areas. In addition, with smartphones, the necessary hardware to receive these signals is widespread. Although such signals are not designed solely for localization purposes, the information they reveal regarding the location of a receiver can be exploited [DSR12].

Indoor scenarios have often been considered difficult for wireless localization due to multipath propagation [YSG09]. A transmitted radio signal is reflected at walls and scattered and diffracted at objects. Therefore, a sum of signal components that have traveled along different propagation paths arrives at the receiver. A signal component is either the line-of-sight (LoS) component or a multipath component (MPC). The LoS component describes the signal component traveling on the direct propagation path from the transmitter to the receiver. All other signal components are MPCs. Using propagation delay based localization methods, for example, multipath propagation causes a bias in the estimate of the propagation time of the LoS component from a transmitter to a receiver. Consequently, the estimated location of the receiver is biased as well.

The term user may refer to a mobile user or the actual receiver a user is equipped with in the following.

1.2 Multipath Assisted Positioning

With multipath assisted positioning, a novel approach that exploits the spatial information contained in MPCs for localization has emerged. It regards each MPC as a LoS signal from a virtual transmitter as exemplarily depicted in Fig. 1.1. Omitting the LoS path from the physical transmitter Tx to the user, a MPC arrives at the user after being reflected at a wall. This MPC can be regarded as a LoS signal from the virtual transmitter vTx, which is located at the location of the physical transmitter mirrored at the wall. A virtual transmitter

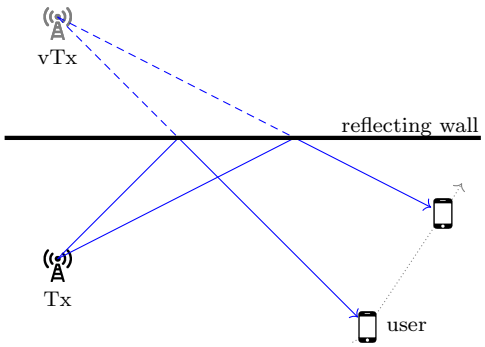


Figure 1.1: In multipath assisted positioning, the MPC arriving at the user after being reflected at the wall is interpreted as a LoS signal from the virtual transmitter vTx.

has no correspondence in the physical world. It merely indicates the apparent origin of a MPC. Each MPC detected by a receiver can be interpreted as a signal from a virtual transmitter in a LoS condition.

If the locations of the physical transmitter and structures in the environment such as reflecting walls or scattering and diffracting objects are known, the locations of the virtual transmitters can be calculated. Structures in the environment may be known from a floor plan, for example. Using virtual transmitters, localizing a user might be possible with only one single physical transmitter. Multipath assisted positioning approaches that assume the environment to be known have been presented for UWB [MGW10; Mei+13; Lei+14] or radar [Set+12] signals, for example. In [NK17], a network of physical transmitters is assumed to be available for multipath assisted positioning. Even with knowledge of a floor plan, the associations of received signal components to physical and virtual transmitters need to be established [OSD15; Lei+16].

In a general setting, information on the environment is not available, and the location of the physical transmitter is unknown. However, the locations of the physical and the virtual transmitters can be estimated jointly with the location of a user with simultaneous localization and mapping (SLAM) [DB06]. In SLAM terms, the user is localized simultaneously with mapping the states of physical and virtual transmitters. A transmitter state comprises both the location of a transmitter and its clock offset towards the user.

Multipath assisted positioning algorithms using SLAM without prior information on the environment have been presented in [Wit+16; Wym+18; Kim+18; Sha+18; Yas+18; Men+19b; Men+19a] for fifth generation (5G) and mmWave systems. UWB systems have been considered in [KAT13; Lei+17], for example. The authors of [Lei+17; Lei+19] focus on algorithms using factor graphs in multipath assisted positioning.

Theoretical bounds for multipath assisted positioning have been provided in [SW09; WM12; SD13; Lei+15; Gen+16a], for example.

With Channel-SLAM, the authors of [GJD13b] and [Gen+16a] have introduced a SLAM based multipath assisted positioning algorithm that does not require prior knowledge of the environment. Channel-SLAM does not differentiate between the LoS component and MPCs. Instead, every signal component is regarded as a LoS signal from a transmitter.

1.3 Goal of the Thesis

Channel-SLAM has been designed for estimating the location of a single user. In many indoor scenarios such as malls, museums or public buildings, a high fluctuation of users can be expected. For each user in a scenario, Channel-SLAM estimates the transmitter states from scratch. Thus, a transmitter's state estimate tends to have a high uncertainty upon initialization, leading to a high uncertainty about the user location, a long convergence time and a limited localization performance.

The goal of this thesis is to exploit the so far unused potential of cooperation in Channel-SLAM. Users can cooperate by exchanging maps of estimated transmitter states to avoid the high uncertainty about the transmitter states when the transmitters are initialized. If a user moves in a scenario with a map obtained from a different user, the information in such a map can be used as prior information. The localization performance is expected to improve and the convergence time to decrease at the same time. In addition, the user can update and improve the information on transmitter states in the prior map. Finally, the updated map can be handed over to the next user. With more and more users contributing to the map, the map becomes more complete in terms of number of transmitters and convergence of transmitter state estimates. In turn, a better prior map improves the localization performance of the users.

The effect of cooperation is exemplarily illustrated in Fig. 1.2. It shows the same scenario as Fig. 1.1, but with a second user and ellipses indicating the uncertainty about the users' locations. The first user simultaneously estimates their location and the state of the transmitters while traveling through the scenario. The uncertainty about the user location drawn in blue decreases only slowly, since the initial uncertainty about the transmitters' states is high. When the first user hands over a prior map of the transmitter state estimates to the second user, the initial uncertainty about the transmitters' states and consequently also about the location of the second user is considerably lower. For simplicity, the uncertainty about the transmitter states is not drawn in Fig. 1.2.

Though, Channel-SLAM is only a relative localization system. The user location is estimated relative to the locations of the transmitters in a local coordinate system that the user creates. In general, the initial position of the user is unknown, and no reference to an absolute coordinate system is known from GNSSs or other localization systems. In an indoor scenario, for example, a user may enter a building from an underground parking garage or a subway station, lose location information in a building, or turn on localization functionality on a smartphone only when in the building.

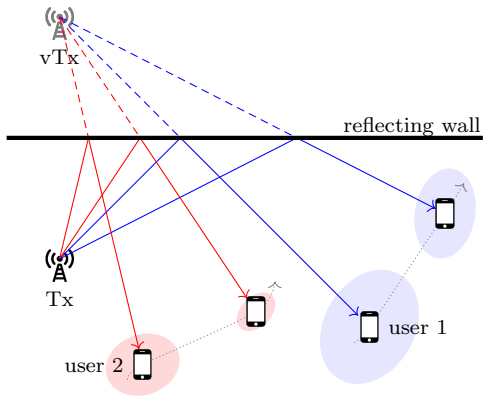
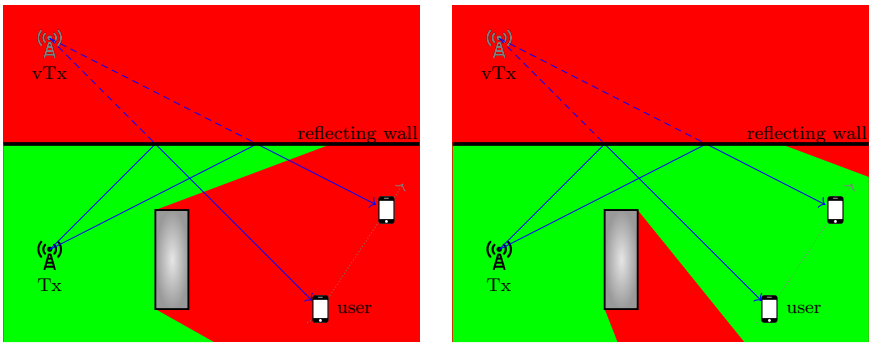


Figure 1.2: The ellipses around the users indicate the uncertainty about their locations. For the first user, the uncertainty about their location is high and the convergence time is long. If the first user shares a map of the transmitters' state estimates with the second user, the initial uncertainty about the transmitters' states and thus about the second user's location is decreased considerably.

When a prior map is handed from one user to another, there are two unknowns that need to be estimated to be able to use this map. First, the coordinate system of the prior map and the coordinate system of the user receiving the prior map are related by a rotation and a two-dimensional translation, leading to three unknown transformation parameters. Second, the correspondences among transmitters in the prior map and transmitters observed by the user receiving the prior map are unknown. It is not clear which of the transmitters observed by the user correspond to which transmitters in the prior map, or if there are such correspondences in the first place.

A prior map contains only estimated locations and clock offsets of transmitters. These estimates are the basis for *map matching*. We define the term map matching as the estimation of the transmitter correspondences and the transformation parameters relating the coordinate systems. In particular in scenarios with only one physical transmitter, ambiguities in map matching occur if only few transmitters have been observed by the user and many transmitters are in the prior map. In this case, it is likely that the arrangement of transmitter locations observed by the user can be found in the prior map multiple times. Likewise, ambiguities in map matching occur if many transmitters have been observed by the user and only few transmitters are in the prior map.

The fundamental problem of map matching is the lack of *diversity* among transmitters. Diversity refers to how easy or hard it is to differentiate among transmitters. Since a physical transmitter and the virtual transmitters corresponding to MPCs transmit the very same signal that is merely reflected and scattered in the environment, it is hard to differentiate among these transmitters, and consequently there is only little diversity among them. Thus, it is hard to associate signal components with transmitters. Likewise, it is hard to find



(a) (b)

Figure 1.3: In (a) and (b), visibility areas of the transmitters Tx and vTx, respectively, are shown. In areas colored green, the respective transmitter is visible, while in red areas it is not.

transmitter correspondences in map matching.

In order to increase the transmitters' diversity, a transmitter state can be augmented by additional information. Within this thesis, the transmitter state is augmented by the information from which locations a transmitter is visible or not. A transmitter is visible from a certain location, if a user can receive the transmit signal in a LoS condition. Following the idea of multipath assisted positioning, each MPC is a LoS signal from a different virtual transmitter. Since different transmitters are visible from different locations, visibility information increases the diversity among transmitters by adding characteristics to transmitters that can help differentiating among them.

Fig. 1.3 visualizes the idea of visibilities of transmitters. It depicts the same scenario as Fig. 1.1, but with an obstacle blocking the transmit signal of the physical transmitter. The visibility areas of the physical transmitter are depicted in Fig. 1.3 (a), and of the virtual transmitter vTx in Fig. 1.3 (b). In green areas, the corresponding transmitter is visible, whereas in red areas, it is not. In particular, at both user positions, the physical transmitter is not visible, whereas the virtual transmitter vTx is.

1.4 Major Contributions

While Channel-SLAM is a promising localization scheme in GNSS denied multipath scenarios if only few physical radio transmitters are available, it targets only single users. In this thesis, methods are developed to expand the existing Channel-SLAM algorithm such that information on estimated transmitter states can be exchanged and exploited to increase the localization accuracy.

A fundamental problem in Channel-SLAM is the lack of diversity among

virtual transmitters. In particular, exchanging maps of estimated transmitter states requires diversity among transmitters for a robust map matching. A major contribution of this thesis is the inclusion of visibility information in Channel-SLAM as a requirement for a reliable cooperation in terms of exchanging maps among users. No additional hardware or infrastructure is necessary to exploit the information on transmitter visibilities. Visibility information is incorporated in the following aspects of Channel-SLAM.

- **New tracking filters using visibility information**
  If a user comes across a location where they had been before, the same set of transmitters that had been visible before is supposed to be visible again. Detecting such a *loop closure* is an important aspect in SLAM schemes. Information on transmitter visibility is used for detecting loop closures in Channel-SLAM and thus increases the localization accuracy. Two new tracking filters are derived to estimate and exploit transmitter visibility information.

- **Data association**
  Data association in multipath assisted positioning describes the association among signal components, i.e., transmitters, over time. Finding associations among transmitters across time is a crucial issue for long term robust SLAM. The lack of diversity among transmitters makes data association a particularly difficult problem in multipath assisted positioning schemes. Augmenting transmitter states by visibility information makes data association much more robust against false associations. A data association scheme using transmitter visibility information is derived for the new tracking filter. It increases the localization performance and in particular the robustness of Channel-SLAM.

- **Cooperation by exchanging maps**
  The lack of diversity among transmitters makes map matching unreliable, as ambiguities are likely to arise if only one physical transmitter is available and certain geometries in the scenario exist. However, with information about the visibility of transmitters, such ambiguities may be resolved. A map matching algorithm is developed that relies on both the transmitter states and visibility information as estimated by the new tracking filter, increasing the robustness of map matching considerably. The ability to exchange prior maps of transmitter states among users for cooperation is a fundamental change in the philosophy of state-of-the-art Channel-SLAM. It allows for a considerably better localization performance and simultaneously decreases the complexity.

Thus, including visibility information does improve the performance of single user Channel-SLAM with a new tracking filter and a reliable data association scheme. Even more important, it enables exploiting the unused potential of cooperation among users.

1.5 Structure of the Thesis

This thesis comprises eight chapters, which are structured as follows.

Fundamental concepts of wireless localization and estimation are introduced in Chapter 2. It focuses on principles that are used in later parts of the thesis.

Chapter 3 presents the idea of multipath assisted positioning and Channel-SLAM as one multipath assisted positioning algorithm in detail. It mainly relies on work published in [Gen+16a] and [Gen18].

In Chapter 4, visibility information on transmitters is introduced. Afterward, a new Rao-Blackwellized particle filter is derived for Channel-SLAM that includes such visibility information. The chapter concludes with the derivation of a reliable data association method based on transmitter states and visibilities for this new filter. The contents of this chapter have been presented in [Ulm+17a; Ulm+17b; UGD19; Ulm+20].

Cooperation among users by exchanging maps of transmitter states and visibilities is explained in Chapter 5. It explains how such maps are created, shared and merged. Parts of this chapter have been published in [ULG18; UG18; UG19; UGD20].

As the transmitter states are estimated by particle filters in Channel-SLAM, the communication load is large when prior maps are exchanged. In Chapter 6, a new tracking filter is derived where the transmitter states are represented by Gaussian mixture models to decrease the communication load. A data association scheme for the filter is derived as well. The contents of the chapter have partly been presented in [Ulm+18].

In Chapter 7, the performance of Channel-SLAM with exchanging transmitter maps among users is analyzed by means of simulations in an indoor scenario with respect to several parameters.

Finally, concluding remarks are made in Chapter 8.

Chapter 2

Concepts of Wireless Localization and Estimation

The estimation of a user's position can be performed with a wide range of sensors and corresponding principles. Each of them have specific strengths and weaknesses, making them suitable to certain scenarios and unfavorable to others. Reasons to decide for a certain localization system include the localization performance, the size, cost and availability of the sensors, necessary or available infrastructure, the reliability of the system, or environmental conditions.

Inertial sensors for example measure accelerations, and do therefore not require an infrastructure of any kind. Since the acceleration measurements need to be integrated twice to obtain a position, errors accumulate and cause large biases in the long term. Similar considerations hold for gyroscopes, which measure turn rates of a device [TWW04].

On mobile platforms, odometry measurements can be used to estimate a position relative to a starting location. Such measurements can be obtained from rotary encoders on wheels, including the steering wheel [DCC03], or cameras [SF11], for example.

With an approach called *fingerprinting*, databases of features in the environment are created. When a user observes certain features, they can be matched against the database to localize the user. Features can be images taken with cameras, for example. Different camera models, exposure times, weather and daytime conditions, or changes in the environment over time complicate the matching process. Furthermore, large databases need to be hold available by or communicated to the user [BOO08].

Magnetic field sensors allow localization by detecting certain patterns in the magnetic field of the earth, or changes in magnetic patterns. They may be prone to interfering magnetic fields from non-stationary objects of magnetic materials or power supply systems, for example [SSR10].

The authors of [Mül+17] have proposed an indoor localization system that is based on electric noses. Electric noses measure gas concentrations in the atmosphere. Based on different concentrations of certain gases, the location of

a user equipped with an electric nose may be determined.

In general, any kind of sensors can be combined in a localization system, and their information fused to create hybrid systems. The strength of one system may compensate the weaknesses of another, and systems therefore complement each other. Combining information from different sensors optimally is a challenging task, as a common system architecture needs to be found [Kle04; KS11; Gro13].

Subsequently, we focus on localization with radio frequency signals. From such signals, position related parameters can be inferred. In Section 2.1, fundamental principles of wireless localization are explained. Section 2.2 presents how a user's position can be estimated and tracked over time in a Bayesian sense.

2.1 Wireless Localization Principles

Wireless localization approaches use radio frequency signals for localization. Such signals often stem from systems that are deployed for the sole purpose of localization. A famous example for such systems are GNSSs such as the American Global Positioning System (GPS) or the European Galileo system. Satellites continuously transmit dedicated signals that users on earth can receive and use to estimate their location. On a smaller scale, ultra-wideband tags or Bluetooth beacons can be deployed on a room or building scale.

However, there are plenty of other sources of radio frequency signals which can be used for localization, such as television signals from satellites or terrestrial transmitters, WLAN or cellular signals, for example. Such signals have in common that they are not designed for the purpose of localizing a user, but they may still be utilized for localization, and are therefore called signals of opportunity (SoOs) [DSR12].

Wireless localization approaches with radio frequency signals can be grouped into two categories. The first category uses fingerprinting techniques, while the second category relies on estimating location related parameters of the received signals.

The first category is based on creating a map or database of signal parameters that have been recorded or calculated prior to the actual position estimation process. Typically, measurements of signal parameters are taken at many points of known locations in the scenario of interest in an offline phase. In this way, a map of so called fingerprints, or training data, is created. Such methods are often called fingerprinting. When a user moves through a scenario in the online phase, they take measurements and match them against the mapped fingerprints to derive their position. Fingerprint types that are widely used are measurements of the received signal strength (RSS), or, in more advanced systems, snapshots of the channel impulse response (CIR). One main challenge of fingerprinting are short- or long-term changes in the environment by new objects, removal of other objects, or people. Such changes lead to different attenuations, reflections and scattering of the signals, and maps of fingerprints becoming outdated. Recent research aims to reduce the number of fingerprints needed to update a

fingerprint map [HC16]. Further more advanced techniques and recent developments include the improvement of fingerprinting models [BK12], exploiting the correlations between signals and structures in the indoor environment, and the calibration of different receiving devices, for example [HC16].

The second category is typically implemented in two stages. In the first stage, position related parameters of the received wireless signal are estimated. Examples for such parameters are the delay or time of arrival (ToA), angle of arrival (AoA), phase, or signal strength. The estimates are used in the second stage to obtain a localization solution.

In the following, we focus on localization principles based on estimated ToAs and AoAs of received signals. The position of the user is denoted by $\boldsymbol{p}_u = [x_u \quad y_u]^T$, and the location of the j^{th} node of a network by $\boldsymbol{p}^{<j>} = [x^{<j>} \quad y^{<j>}]^T$. Such nodes are transmitters or transceivers, which are able to send and receive signals to and from a user.

2.1.1 Time of Arrival Measurements

The expression ToA denotes a point in time when a signal is received. In the following, we will use the expressions ToA and signal delay synonymously with time of flight, i.e., the time a signal takes to travel from a network node to a user or vice versa. The estimation of the ToA of a signal thus yields information about the distance between a user and a network node. If such ToA estimates are obtained for a sufficient number of nodes with known location in the network, the position of the user can be calculated. A simplified model of the received signal $r(t)$ over time t is expressed as

$$r(t) = s(t - \tau(t)) + n(t), \tag{2.1}$$

where the transmit signal is denoted by $s(t)$, $\tau(t)$ is the ToA, and $n(t)$ is a stochastic noise process. The estimation of the actual ToA can be performed by means of a correlator [YSG09] or matched filters [Tur60], for example. When the nodes are time synchronized among themselves, but not to the user, the clock offset e of the user relative to the network needs to be estimated in addition. The function $h_{\text{ToA}}(\cdot)$ relating the user position $\boldsymbol{p}_u$, the location $\boldsymbol{p}^{<j>}$ of the j^{th} node and the corresponding ToA $\tau^{<j>}$ can be expressed by

$$\tau^{<j>} = h_{\text{ToA}}(\boldsymbol{p}_u, \boldsymbol{p}^{<j>}, e) = \frac{1}{c_0}\|\boldsymbol{p}_u - \boldsymbol{p}^{<j>}\| - e, \tag{2.2}$$

where c_0 denotes the speed of light. The ToA may be expressed in time domain as τ or in distance domain as $d = c_0\tau$.

If a sufficient number of measurements to different nodes is available, the user location can be estimated. Fig. 2.1 depicts an example for ToA localization with three transmitting nodes under the simplified assumption of no measurement errors. The three ToAs $\tau^{<j>}$ between the user and the three network nodes correspond to the three distances $d^{<j>} = c_0\tau^{<j>}$ between the user and the nodes, where j is the index of a node. Consequently, the user position is the intersection of the three circles with radii $d^{<j>}$.

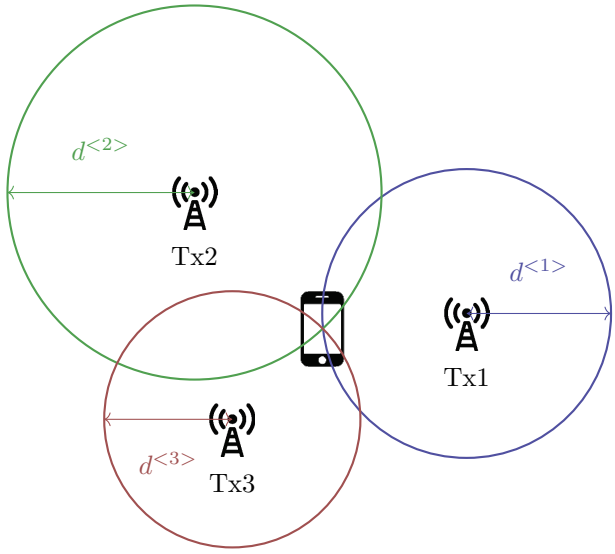


Figure 2.1: Around each of the three network nodes Tx1, Tx2 and Tx3, a circle is drawn whose radius corresponds to the distance $d^{<j>}$ between the user and the corresponding transmitter. This distance is obtained by from the corresponding ToA measurement. The three circles intersect in the user location.

In an n-dimensional scenario where the user is perfectly time synchronized to the network, at least $n + 1$ nodes are necessary to uniquely determine the location of the user. In the two-dimensional scenario in Fig. 2.1, for example, all three nodes are necessary. For each additional parameter to estimate, such as user clock offset, a ToA measurement to one more node is necessary.

An example for ToA based localization systems are GNSSs [ME11]. The GNSS satellites are time synchronized by atomic clocks and orbit the earth on known trajectories. The satellites continuously transmit navigation messages. Users jointly estimate their own positions and clock offsets relative to the satellite network.

A distance can also be estimated with two-way ranging between a user and a transceiver. A signal from the user is received by a network node and immediately transmitted back, or vice versa. From the time between transmission of the signal and reception of the response, the user can estimate the distance to the node. In such a case, the unknown clock offset between user and the network does not influence the distance estimate. Such a protocol is implemented in the IEEE 802.11 WLAN standard [IEE16], for example. This specific protocol relies on a two-way ranging process between a router and mobile terminal. Users can thus obtain range estimates to a set of WLAN routers and estimate their own location.

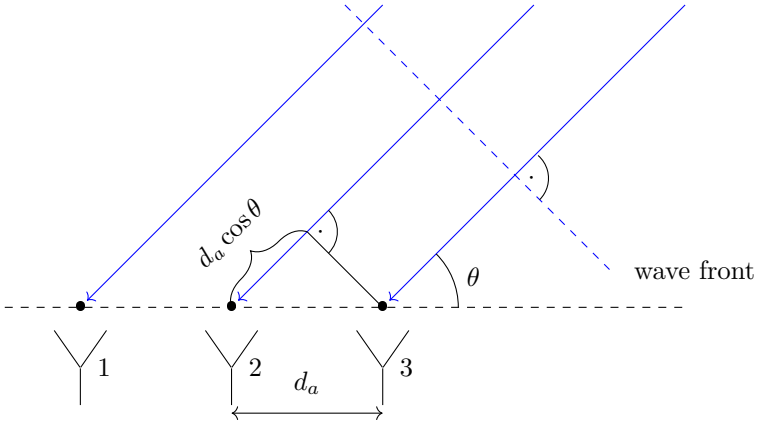


Figure 2.2: A wave front arrives at an antenna array of three elements with element spacing d_a under an angle θ.

2.1.2 Angle of Arrival Measurements

Information about the AoA of a received signal is information about the direction the signal is coming from. Typically, antenna arrays are used to obtain information about the AoA. The delay of a signal at an antenna element is specific to the location of each antenna element in the array. If the geometry of the array is known, the differences in these delays can be used to calculate the AoA of the signal relative to the array orientation. Fig. 2.2 shows a linear antenna array with three elements as an example. The waves arrive at the antenna elements at an angle θ. With an antenna element spacing of d_a, the delay difference among neighboring antennas for an incoming wavefront is ${}^{d_a}/_{c_0} \cos\theta$. For narrow-band signals, the differences in the delays correspond to differences in the phase of the signal.

Fig. 2.3 illustrates localization with AoA information in absence of measurement errors. The AoAs of signals of two nodes Tx1 and Tx2, $\theta^{<1>}$ and $\theta^{<2>}$, respectively, are sufficient to determine the position of the user in a two-dimensional scenario, if the orientation $\boldsymbol{u}$ of the antenna array is known. If the orientation is unknown, it has to be estimated as an additional parameter, in which case one more AoA measurement is required. Since the AoA information is independent from time, synchronization among the network nodes themselves or network nodes and the user is not necessary.

A simplified model of the received signal for the m^{th} antenna element in the array is described by [RRL99]

$$r_m(t) = a_m(\theta) s(t - \tau_m(t)) + n(t), \tag{2.3}$$

where θ is the AoA of the signal and $n(t)$ a stochastic noise process. The scalar $a_m(\theta)$ is the response of the m^{th} antenna element dependent on the AoA θ, and $\tau_m(t)$ the ToA of the received signal at the m^{th} antenna element.

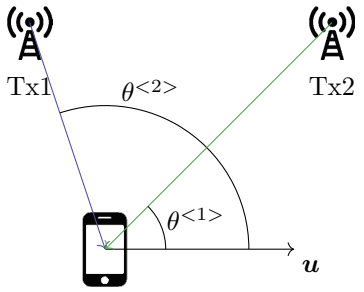


Figure 2.3: The signals from two network nodes Tx1 and Tx2 arrive at the user under angles $\theta^{<1>}$ and $\theta^{<2>}$, respectively, relative to the user orientation $\boldsymbol{u}$. The user is located at the intersection of the corresponding lines.

The function linking the AoA measurement $\theta^{<j>}$ to the user position and the location of the j^{th} node is expressed by

$$\theta^{<j>} = h_{\text{AoA}}(\boldsymbol{p}_u, \boldsymbol{p}^{<j>}) = \text{atan2}\left(y_u - y^{<j>}, x_u - x^{<j>}\right), \tag{2.4}$$

where $\text{atan2}\,(y, x)$ is the four-quadrant inverse tangent function. It returns the unique counter-clockwise angle between the positive x-axis and the line connecting the origin with the point given by the coordinates (x, y).

2.1.3 Performance Issues

In reality, the parameter estimates such as ToA and AoA are subject to various sources of error degrading the localization performance. The most prominent source is thermal noise affecting the transmitter and receiver hardware. Three more sources of error are considered in the following, namely multipath propagation, non-line-of-sight (NLoS) propagation and the geometrical dilution of precision (GDoP), as they play an important role in the following chapters.

2.1.3.1 Multipath Propagation

Multipath propagation is illustrated in Fig. 2.4, where the signal from the transmitter Tx is received by the user via different propagation paths due to reflections and scattering in the surroundings. The LoS component arrives at the receiver on the direct propagation path. After undergoing a reflection at a wall, a second signal component arrives at the receiver as the first NLoS component, or MPC. A second and a third MPC arrive at the receiver after being scattered at point scatterers. Each signal component arrives at the receiver with its own signal parameters, such as amplitude, phase, ToA and AoA.

Although additive noise is always present, we consider a received signal without noise in the following considerations for clarity. Furthermore, dense multipath components (DMCs) [Ric05] are neglected. DMCs are the MPCs in a radio channel that cannot be resolved by the measurement aperture.

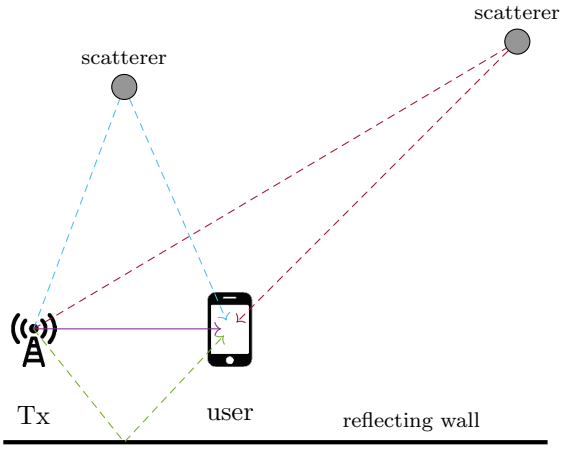


Figure 2.4: The transmit signal from the network node Tx arrives at the user via different propagation paths. The direct path is drawn solid, whereas reflected and scattered MPCs are drawn with dashed lines.

The CIR $h(t)$ of a simple static multipath channel can then be expressed as a sum of Dirac functions

$$h(t) = \sum_j \alpha^{<j>} \delta\left(t - \tau^{<j>}\right), \tag{2.5}$$

where $\alpha^{<j>}$ and $\tau^{<j>}$ are the complex amplitude and ToA, respectively, of the j^{th} signal component. The received signal $r(t)$ is the convolution of the transmit signal with the CIR. The received signal is therefore a sum of the transmit signal $s(t)$ arriving at the receiver with different parameters such as ToAs, AoAs, amplitudes and phases. It is expressed as

$$r(t) = \sum_j \alpha^{<j>} s\left(t - \tau^{<j>}\right). \tag{2.6}$$

In real systems, the signal bandwidth is limited, though. The CIR and a superimposed received signal with limited bandwidth for the scenario in Fig. 2.4 are plotted in Fig. 2.5. The CIR consists of the four Dirac functions shown with diamond shaped markers. The first Dirac function with ToA τ_1 corresponds to the LoS component. The second Dirac function with ToA τ_2 corresponds to the MPC reflected at the wall, while the two Dirac functions with delays τ_3 and τ_4 correspond to the MPCs scattered at the two point scatterers in the scenario in Fig. 2.4.

The transmit signal considered in Fig. 2.5 is of rectangular shape in frequency domain, which corresponds in time domain to the sinc function

$$\text{sinc}(Bt) = \frac{\sin(Bt)}{Bt}, \tag{2.7}$$

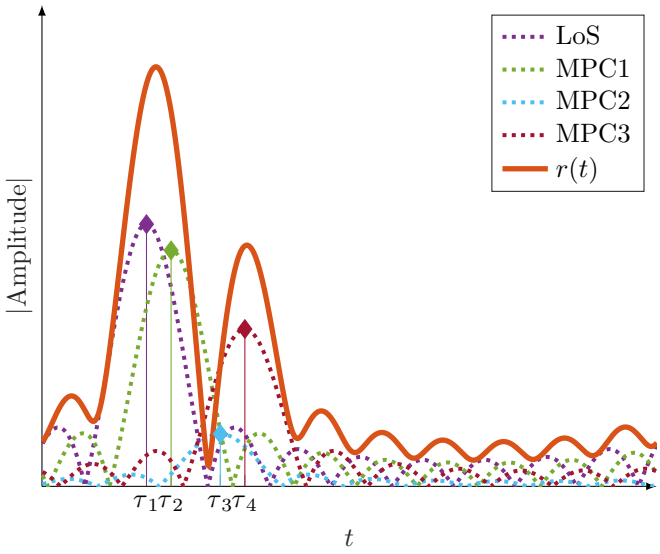


Figure 2.5: Different signal components correspond to different sinc functions. The absolute value of the sum of the sinc functions is plotted as a thick red line.

where B corresponds to the bandwidth.

In Fig. 2.5, the absolute values of the contributions to the received signal corresponding to the single propagation paths are drawn in time domain with dotted lines. Thus, each dotted line is the absolute value of a scaled version of the transmit signal shifted in time and phase. The colors of the sinc functions correspond to the colors of the propagation paths in Fig. 2.4. The absolute value of the received signal $r(t)$, which is the sum of the four signal components' contributions, is plotted in red with the solid line.

The principle of estimating the ToA of a signal often relies on finding the maximum of the first significant peak in the received signal. The argument of the sinc function in Eq. (2.7) is proportional to the bandwidth used. A larger signal bandwidth leads to a more narrow sinc function. The higher the bandwidth, the sharper are the single sinc components, and the more likely is it that no other signal component's sinc curve has influence on the ToA estimate. In such a case, a ToA estimator would be unbiased. In contrast, if the bandwidth of the transmit signal is small, the influence of signal components on each other is significant and thus a biased estimate is more likely.

Fig. 2.6 illustrates such a case. The received signal is plotted for the delays from Fig. 2.5 for different bandwidths. The energies of the received signals are not equal for the sake of clarity of the figure. Denoting the bandwidth in Fig. 2.5 by B_0, the considered bandwidths are $0.5B_0$, B_0, and $5B_0$. It becomes evident that the second signal component shifts the maxima of the curves further to the right away from τ_1. The shift increases as the bandwidth decreases. The excess propagation delay for the blue curve with $0.5B_0$, i.e., the difference between

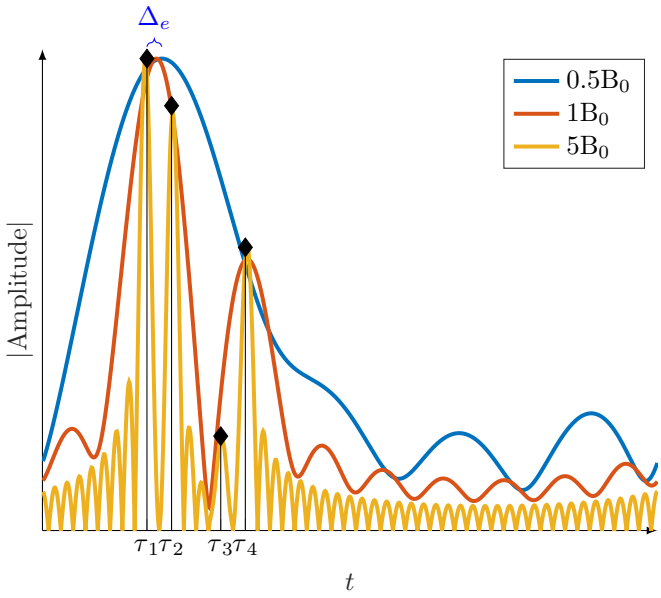


Figure 2.6: The received signal is plotted for three different bandwidths. The maximum amplitudes of the curves are set equal for clarity.

the delay of the maximum of the received signal and τ_1, is labeled Δ_e. If the bandwidth increases towards infinity, this propagation delay decreases towards zero.

Furthermore, thermal noise on the transmit and receiver hardware affects the signals as well, leading to additional errors on ToA estimates. The influence of noise is likely to be greater the smaller the signal bandwidth is, as the peaks of the sinc functions are sharper for a higher bandwidth.

Hence, multipath propagation causes an excess propagation delay which biases the ToA estimate. It can be a significant source of error in localization systems, particularly in urban or indoor scenarios [ZGL19; del+18]. In the literature, various measures to cope with multipath propagation are proposed. One measure is the proper modeling of the propagation conditions. If a statistical model of the scenario, such as indoor or urban, is known, multipath propagation errors can be reduced [YSG09].

A different approach is to use advanced signal processing methods. Some methods try to estimate the true CIR between a node and the user, for example with Space-Alternating Generalized Expectation-Maximization (SAGE) [Fle+99] or Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT) [PRK85]. If the CIR is known, the influence of MPCs on the estimate of the LoS component can be mitigated. In [LK06], for example, the authors use a maximum likelihood (ML) estimator in a delay-locked loop to estimate the actual ToA of the signal. Such advanced signal processing methods come at the cost of a higher computational complexity, though.

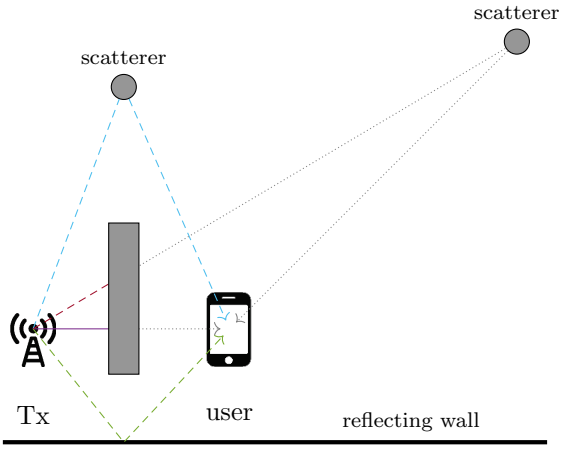


Figure 2.7: The direct path between the network node Tx and the user is blocked. The transmit signal from the network node Tx arrives at the user via two different propagation paths due to a reflection and scattering at objects in the environment. Blocked propagation paths are indicated by the gray dotted lines.

2.1.3.2 Non-Line-of-Sight Propagation

In complex environments, there is often no LoS condition between a network node and the user, as signals may be blocked by obstacles and objects like walls. However, a transmit signal from a network node might still arrive at the user via other propagation paths. Such a situation is depicted in Fig. 2.7, which shows the same scenario as Fig. 2.4, but with an obstacle blocking the LoS component and the MPC corresponding to the propagation path via a scatterer. The blocked signal components are grayed out. The situation where no LoS component is received as in Fig. 2.7 is referred to as NLoS propagation. The first signal component arrives at the user after being reflected at the wall. The user may falsely interpret this MPC as the LoS signal. In this case, the parameters of the received signal estimated at the user are biased. On the one hand, the ToA estimate is biased by a positive offset, the excess propagation delay. On the other hand, the AoA of the first MPC received by the user is different from the blocked LoS component. In addition, a lower signal-to-noise ratio (SNR) of the reflected signal compared to a LoS signal further deteriorates the localization performance [QK02]. Hence, NLoS propagation can be one of the most significant sources of error [SR96].

There are several techniques to deal with NLoS propagation. A first step is to detect when the LoS signal between node and user is lost, and thus to identify a NLoS propagation condition. One approach is to take a hypothesis test of current range estimates against the history of estimates. In addition, the standard deviation of the measurement noise can be taken into account in

such a test [WH96]. Other approaches include distribution tests, where statistical properties of the received signal are evaluated to differentiate between LoS and NLoS propagation [YSG09]. When NLoS propagation is detected, the corresponding measurement may be corrected by estimating the excess propagation delay, or excluded from the set of available measurements if the system is overdetermined.

2.1.3.3 Geometric Dilution of Precision

Multipath propagation and NLoS propagation affect the estimates of parameters of signal components. Independently from the quality of these estimates, the geometry of the scenario influences the performance of localization algorithms. In Fig. 2.8, there are two situations with two network nodes Tx1 and Tx2 and one user at an unknown position indicated by the red diamond. In Figs. 2.8 (a) and (b), the user and the node Tx1 are at the same locations, while the location of node Tx2 is different. The locations of both nodes are known. The ToA estimates to the two transmitters correspond to the distance estimates $d^{<1>}$ and $d^{<2>}$. For the sake of clarity, the distance estimates are offset by a maximum of $\pm\delta_u$. The user must therefore be on both the blue annulus around Tx1 and the green annulus around Tx2. The area where the two annuli overlap and thus the uncertainty about the user location is significantly larger in Fig. 2.8 (b) compared to the overlapping area in Fig. 2.8 (a). The uncertainty due to the geometry of the scenario is often denoted by the term geometrical dilution of precision.

Denote the angle between the two lines connecting the user with the two network nodes as in Fig. 2.8 by θ. It is intuitive that the geometrical dilution of precision of the user location is related to θ. In fact, it is proportional to the inverse of $\sin\theta$ [YSG09]. As θ gets closer to an uneven multiple of $\pi/2$ as in Fig. 2.8 (a), the geometrical dilution of precision shrinks. If the angle gets close to a multiple of π as in Fig. 2.8 (b), it increases.

More insight into the geometrical dilution of precision is given in [Kel03] and [YSG09]. The relationship between the geometrical dilution of precision and the Fisher information is presented in [CA94]. The concept of geometrical dilution of precision has mostly been investigated for GNSSs with respect to satellite constellations [PS96; KH05].

2.2 Estimation and Tracking Principles

When estimating the state of a parameter, it is often not possible to measure the state directly, but only to measure quantities that are somehow related to the state. For example, in Eq. (2.2) and Eq. (2.4), it is not the position of a user that is measured, but quantities such as the ToA or AoA of the received signal from which this position can be inferred. These measurements, or observations, are typically noisy. Such noise is due to the use of imperfect sensors, thermal noise, or measurement specific errors, for example. In terms of ToA and AoA estimation, such specific errors could be multipath or NLoS propagation

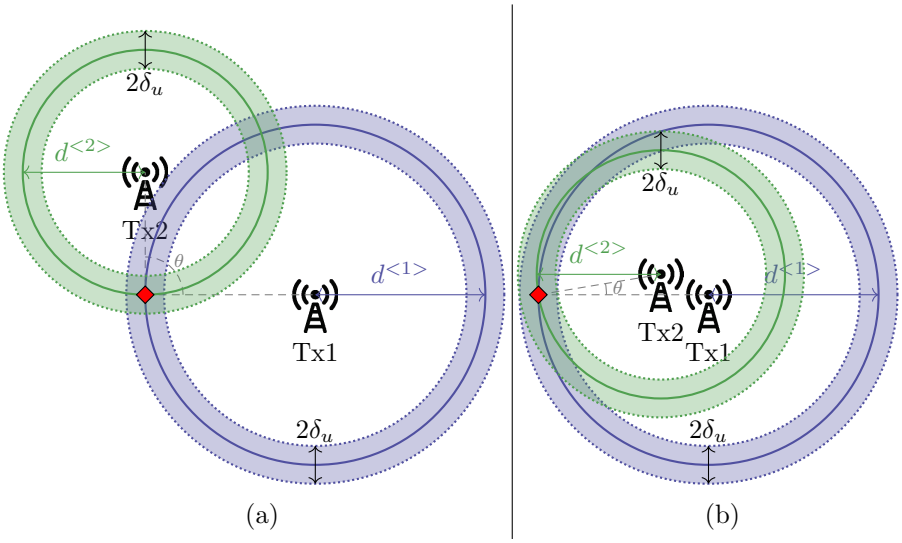


Figure 2.8: The network nodes Tx1 and Tx2 are arranged in different ways in (a) and (b). The user position is indicated by the red diamond. The delay offset measurements are affected by noise, leading to an uncertainty of $\pm\delta_u$ indicated by the transparent annuli around the nodes. The overlapping area of these annuli is in (a) much smaller than in (b), although there is an ambiguity in the user position in (a).

[YSG09]. Estimators seek to find the optimal estimate for the parameter based on the noisy measurements [BLK04]. Optimality may refer to the maximization of a likelihood or minimization of an error function, for example.

When the parameters to be estimated are not static, but change over time, they are referred to as the state of a system, and the steady estimation of the state as tracking of a dynamic system [BLK04]. If there are correlations between states at subsequent time instants, knowledge from one time instant can be taken into account as prior knowledge in the next time instant. In the following, recursive Bayesian estimation is introduced as a concept of tracking. Afterward, two algorithms implementing discrete-time recursive Bayesian estimation are presented, which are the Kalman filter and the particle filter.

2.2.1 The Bayesian Philosophy

In classical estimation approaches, the parameter or state $\boldsymbol{x}$ to be estimated is a deterministic but unknown constant. The data model is expressed by the known probability density function (PDF)

$$\mathrm{p}\left(\boldsymbol{z};\boldsymbol{x}\right), \tag{2.8}$$

which denotes the PDF of the measurement $\boldsymbol{z}$ parametrized by $\boldsymbol{x}$. In contrast, the idea behind the Bayesian philosophy of estimation is to consider $\boldsymbol{x}$ a realization of a random variable or vector [Kay98]. Thus, the model of interest is the joint PDF of the random variables $\boldsymbol{z}$ and $\boldsymbol{x}$,

$$\mathrm{p}\left(\boldsymbol{z},\boldsymbol{x}\right) = \mathrm{p}\left(\boldsymbol{z}|\boldsymbol{x}\right)\mathrm{p}\left(\boldsymbol{x}\right). \tag{2.9}$$

The fundamental difference to classical approaches is that prior knowledge on the state, namely $\mathrm{p}\left(\boldsymbol{x}\right)$, is incorporated in the estimation process.

An important estimator is the minimum mean square error (MMSE) estimator. It minimizes the expectation value of the mean square error (MSE) between the random variable $\boldsymbol{x}$ and the estimator. In the classical case, this expectation value is minimized with respect to the parametrized PDF $\mathrm{p}\left(\boldsymbol{z};\boldsymbol{x}\right)$. In the Bayesian philosophy, the expectation value is calculated with respect to the joint PDF $\mathrm{p}\left(\boldsymbol{z},\boldsymbol{x}\right)$. This leads to the Bayesian MMSE estimator [Kay98]

$$\hat{\boldsymbol{x}}_{\mathrm{MMSE}} = \arg\min_{\hat{\boldsymbol{x}}} \int \|\boldsymbol{x}-\hat{\boldsymbol{x}}\|^2 \,\mathrm{p}\left(\boldsymbol{x}|\boldsymbol{z}\right)\mathrm{d}\boldsymbol{x} = \int \boldsymbol{x}\,\mathrm{p}\left(\boldsymbol{x}|\boldsymbol{z}\right)\mathrm{d}\boldsymbol{x}. \tag{2.10}$$

2.2.2 Recursive Bayesian Estimation

Recursive Bayesian estimation is a method to recursively estimate the evolution of the dynamic state vector $\boldsymbol{x}$ in the Bayesian philosophy. In particular, the posterior PDF

$$\mathrm{p}\left(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}\right) \tag{2.11}$$

is sought, where $\boldsymbol{x}_k$ is the state vector at time instant k, and $\boldsymbol{z}_{1:k}$ are the measurements from time instants 1 to k.

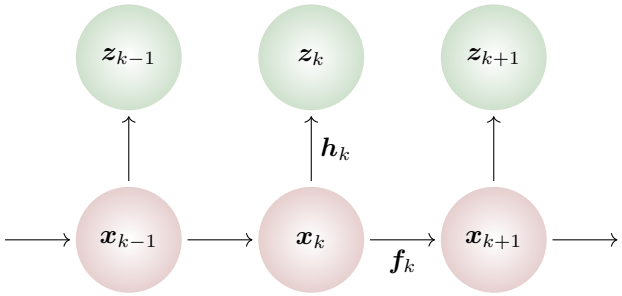


Figure 2.9: The dynamic Bayesian network illustrates the conditional dependencies between the state variable $\boldsymbol{x}$ and the measurements $\boldsymbol{z}$ over time, assuming a Markov process of first order.

Recursive Bayesian estimation works in two stages. In the prediction stage, the evolution of the state from one time instant to another is predicted based on a system model. In the update stage, the predicted state is updated by measurements.

The evolution of the state from one time instant $k-1$ to time instant k is modeled by the system model

$$\boldsymbol{x}_k = \boldsymbol{f}_k\left(\boldsymbol{x}_{k-1}, \boldsymbol{w}_{k-1}\right), \tag{2.12}$$

where the function $\boldsymbol{f}_k(\cdot)$ is assumed to be known. To model arbitrary behavior of the system or an uncertainty in the system dynamics, a sample $\boldsymbol{w}_{k-1}$ of the process noise with known covariance matrix $\boldsymbol{Q}_k$ is introduced. In localization problems, the system model is often referred to as movement model. It describes the stochastic movement of a pedestrian or vehicle, for example, taking into account certain dynamics [Sär13].

With the measurement model

$$\boldsymbol{z}_k = \boldsymbol{h}_k\left(\boldsymbol{x}_k, \boldsymbol{n}_k\right), \tag{2.13}$$

the state is linked to the available measurements at time instant k to update the state estimate. The function $\boldsymbol{h}_k(\cdot)$ is assumed known and the measurement noise sample $\boldsymbol{n}_k$ models uncertainty in the measurement. It is a realization of a measurement noise process with known covariance matrix $\boldsymbol{R}_k$. Examples for simplified measurement models without noise are given by Eq. (2.2) for ToA measurements, and by Eq. (2.4) for AoA measurements.

Fig. 2.9 depicts a simple dynamic Bayesian network, which shows the estimated states, the measurements and their relationships and dependencies over time. In Fig. 2.9, the measurement $\boldsymbol{z}_k$ at time instant k is only dependent on the state $\boldsymbol{x}_k$ at time instant k. Likewise, the state $\boldsymbol{x}_k$ depends only on the previous state $\boldsymbol{x}_{k-1}$. If this previous state is known, $\boldsymbol{x}_k$ becomes conditionally independent from any state before time instant $k-1$. Such a relationship is often assumed and describes a Markov process of order one [CT06]. Mathematically speaking,

$$\mathrm{p}\left(\boldsymbol{x}_k | \boldsymbol{x}_{0:k-1}\right) = \mathrm{p}\left(\boldsymbol{x}_k | \boldsymbol{x}_{k-1}\right), \tag{2.14}$$

where $\boldsymbol{x}_{0:k-1}$ is the history of the states from time instants zero to $k-1$. The system model in Eq. (2.12) is related to the horizontal arrows representing the evolution of the state. Likewise, the measurement model in Eq. (2.13) is related to the vertical arrows representing the link between the state and the measurements.

In a Bayesian perspective, the goal is to recursively calculate the degree of belief in the state $\boldsymbol{x}_k$ at time instant k based on the measurements from time instants 1 to k, $\boldsymbol{z}_{1:k}$. Thus, we seek the posterior distribution $\mathrm{p}\left(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}\right)$ at time instant k. In the prediction step, the prior PDF of the current state $\boldsymbol{x}_k$ given the measurements from time instants one to $k-1$ is calculated by

$$\mathrm{p}\left(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1}\right) = \int \mathrm{p}\left(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}\right) \mathrm{p}\left(\boldsymbol{x}_{k-1}|\boldsymbol{z}_{1:k-1}\right) \mathrm{d}\boldsymbol{x}_{k-1}, \tag{2.15}$$

where we assume a first order Markov process as in Fig. 2.9. Hence, the state at time instant k is independent from the measurements from time instants one to $k-1$, if the state at time instant $k-1$ is known, since the information about the measurements from time instants one to $k-1$ is contained in state at time instant $k-1$. In the update step, the posterior PDF of the current state $\boldsymbol{x}_k$ given the measurements from time instants 1 to k is calculated with Bayes' theorem by

$$\mathrm{p}\left(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}\right) = \frac{\mathrm{p}\left(\boldsymbol{z}_k|\boldsymbol{x}_k\right) \mathrm{p}\left(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1}\right)}{\mathrm{p}\left(\boldsymbol{z}_k|\boldsymbol{z}_{1:k-1}\right)}. \tag{2.16}$$

The denominator in Eq. (2.16) is independent from the state $\boldsymbol{x}_k$ and therefore only a normalizing constant.

The first factor in the integral of Eq. (2.15) relates to the system model in Eq. (2.12), while the second factor is obtained by the update step in Eq. (2.16) at time instant $k-1$. Likewise, the second factor in the nominator Eq. (2.16) is obtained from the previous prediction in Eq. (2.15). The first factor in the nominator is related to the measurement model in Eq. (2.13), and it is referred to as the likelihood function. For a given prior distribution $\mathrm{p}\left(\boldsymbol{x}_0\right)$ of the initial state $\boldsymbol{x}_0$, the state posterior $\mathrm{p}\left(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}\right)$ can in principle be calculated recursively with the above equations.

Tracking a state over time may allow obtaining an estimate in an underdetermined system, if a part of the state to estimate can be assumed static. To illustrate that idea, consider a two-dimensional scenario with two network nodes Tx1 and Tx2 of known locations, and one user with unknown initial location $\boldsymbol{p}_{u,0}$ as depicted in Fig. 2.10.

The user is assumed to be time synchronized to the network. There are four unknowns, namely the two locations $\boldsymbol{p}_{u,0}$ and $\boldsymbol{p}_{u,1}$ of the user in two dimensions each. With only one ToA measurement from one user location to the two network nodes, the user location can not be solved uniquely at one time instant, indicated by the two intersection points of each pair of circles around the nodes. The solid circles intersect at the first user position $\boldsymbol{p}_{u,0}$ and at a different position. Likewise, the dashed circles intersect at the next user position $\boldsymbol{p}_{u,1}$ and at one more point.

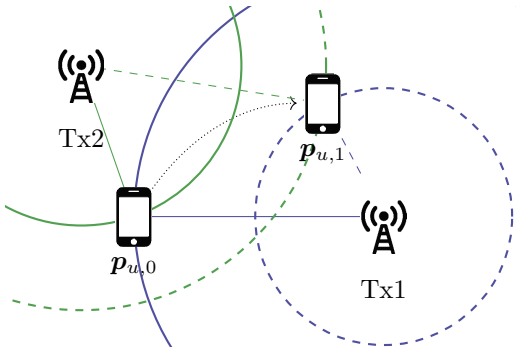


Figure 2.10: The user takes ToA measurements at two different positions $\boldsymbol{p}_{u,0}$ and $\boldsymbol{p}_{u,1}$ at time instants zero and one. The ToAs correspond to the radii of the circles around the network nodes Tx1 and Tx2.

However, the user may move within the network, and take ToA measurements from different locations. The step $\boldsymbol{p}_{u,1} - \boldsymbol{p}_{u,0}$ from one user position to another is indicated by the dotted arrow. Assume that the relative movement of the user and thus the system model as in Eq. (2.12) is known with only some uncertainty, for example by the history of the movement or from the odometry. For each time instant, there are two new ToA measurements, corresponding to two equations linking the unknowns to measured values. Over time, this system becomes overdetermined, and tracking the user position with ToA measurements within the network is possible. As the user moves in the scenario taking ToA measurements to the nodes, the uncertainty about the step vectors and consequently the user positions is reduced over time.

In absence of knowledge of the original state $\boldsymbol{x}_0$, the prior distribution $\mathrm{p}(\boldsymbol{x}_0)$ may be chosen to be uniform. In localization problems, there often is prior knowledge of some kind available. For example, a rough location may be given by the identifier of a received cellular or WLAN signal from a transmitter whose location is known at least to a certain extent.

2.2.3 Optimal and Non-Optimal Estimators

The integral in Eq. (2.15) is in general intractable or does not have a closed-form solution at all, depending on the PDFs in the integral. Nevertheless, in some special cases, closed-form solutions to recursive Bayesian estimation that can be calculated with efficient algorithms exist. For example, under the assumptions that

1. the system and measurement noise terms in Eq. (2.12) and Eq. (2.13), respectively, are drawn from Gaussian distributions,
2. the prior PDF $\mathrm{p}(\boldsymbol{x}_0)$ follows a Gaussian distribution, and
3. the system and measurement models in Eq. (2.12) and Eq. (2.13), respectively, are described by linear functions,

an optimal solution can be obtained with the Kalman filter [Kal60].

The Kalman filter is optimal in an MSE sense, i.e., it minimizes the MSE of the estimated quantities. In absence of the above assumptions regarding linearity and Gaussian noise, non-optimal solutions using approximations exist. In the following, the Kalman filter, the unscented Kalman filter (UKF), the Gaussian sum filter (GSF), and the particle filter, which is such an approximation, are presented.

2.2.3.1 The Kalman Filter

Under the assumption of linear models, a Gaussian prior distribution and Gaussian noise terms as above, all involved PDFs in the recursive Bayesian estimation process are Gaussian distributed. This can be explained by the fact that the conjugate prior of the Gaussian distribution is the Gaussian distribution itself. Hence, updating a prior Gaussian PDF with data which are also Gaussian distributed results again in a Gaussian PDF.

A Gaussian PDF of the state $\boldsymbol{x}$ with dimension $N \times 1$ can be parametrized by a mean $\boldsymbol{\mu_x}$ and a covariance matrix $\boldsymbol{C_x}$. It is denoted by $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu_x}, \boldsymbol{C_x})$ and defined by

$$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu_x}, \boldsymbol{C_x}) = \frac{1}{\sqrt{(2\pi)^N \det \boldsymbol{C_x}}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu_x})^T \boldsymbol{C_x}^{-1}(\boldsymbol{x}-\boldsymbol{\mu_x})}. \tag{2.17}$$

If a scalar value is estimated, both the mean and the covariance involved in the Kalman filter are scalars as well. An intuitive derivation of the Kalman filter for this case is presented in [Far12]. The equations of the Kalman filter are presented in Appendix A.1.

In localization problems, the measurement model is rarely linear. For ToA estimation in more than one dimension as well as for AoA estimation, the models are nonlinear, as can be seen in Eqs. (2.2) and (2.4), respectively. If any model is nonlinear, it can be linearized around the operating point. The extended Kalman filter (EKF) [BLK04] uses a first-order Taylor series expansion for linearization of the system models. However, the extended Kalman filter is not an optimal filter, and its performance depends on the degree of the nonlinearity of the system, which may introduce large errors into the estimates [WM00].

2.2.3.2 The Unscented Kalman Filter

A variant of the Kalman filter is the unscented Kalman filter [JUD95; JU97; WM00], which employs a nonlinear transformation to deal with nonlinear system and measurement models. In particular, it relies on the unscented transform [Jul02]. If a random variable $\boldsymbol{x}$ of dimension N undergoes a nonlinear transformation $\boldsymbol{g}(\cdot)$, the statistics of the resulting random variable $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x})$ can in general not be computed in closed form. The idea of the unscented transform is to propagate a set of deterministically chosen, weighted points, called sigma points, through the nonlinear function. Based on the resulting points, a mean and covariance are approximated for the random variable $\boldsymbol{y}$.

An outcome of the unscented transform are numerical approximations of certain types of integrals. If the integrand is a product of a function $\boldsymbol{g}(\boldsymbol{x})$ and a Gaussian PDF $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_{\boldsymbol{x}}, \boldsymbol{C}_{\boldsymbol{x}})$ with mean $\boldsymbol{\mu}_{\boldsymbol{x}}$ and covariance $\boldsymbol{C}_{\boldsymbol{x}}$, the integral can be approximated by

$$\int \boldsymbol{g}(\boldsymbol{x}) \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_{\boldsymbol{x}}, \boldsymbol{C}_{\boldsymbol{x}}) \, \mathrm{d}\boldsymbol{x} \approx \sum_{m=1}^{N_{\mathrm{sig}}} \omega_m \boldsymbol{g}(\boldsymbol{X}_m), \tag{2.18}$$

where $\boldsymbol{X}_m$ is the m^{th} of the N_{sig} sigma points, and ω_m its associated weight. Valuable insight into sigma point methods and their relation to Gaussian process quadratures can be found in [Sär+16].

In the literature, various sets of sigma points have been proposed [Jul02; AH09]. Sigma points are chosen with the goal to preserve many moments of the original statistics of $\boldsymbol{x}$ through the transform. Since the number of sigma points influences the computational complexity, a small number of points is desirable.

The authors of [JUD95] chose the $N_{\mathrm{sig}} = 2N + 1$ sigma points and their weights for a real number κ as

$$\begin{aligned}
\boldsymbol{X}_0 &= \boldsymbol{\mu}_{\boldsymbol{x}}, & \omega_0 &= \frac{\kappa}{\kappa + N}, \\
\boldsymbol{X}_m &= \boldsymbol{\mu}_{\boldsymbol{x}} + \left(\sqrt{(N+\kappa)\boldsymbol{C}_{\boldsymbol{x}}} \right)_m, & \omega_m &= \frac{1}{2(\kappa + N)}, \\
\boldsymbol{X}_{m+N} &= \boldsymbol{\mu}_{\boldsymbol{x}} - \left(\sqrt{(N+\kappa)\boldsymbol{C}_{\boldsymbol{x}}} \right)_m, & \omega_{m+N} &= \frac{1}{2(\kappa + N)},
\end{aligned} \tag{2.19}$$

for $m = 1, ..., N$. The vector $(\boldsymbol{A})_m$ is the m^{th} column of the matrix $\boldsymbol{A}$, and $(N + \kappa)\boldsymbol{C}_{\boldsymbol{x}}$ is factorized such that

$$(N+\kappa)\boldsymbol{C}_{\boldsymbol{x}} = \sqrt{(N+\kappa)\boldsymbol{C}_{\boldsymbol{x}}} \sqrt{(N+\kappa)\boldsymbol{C}_{\boldsymbol{x}}}^T. \tag{2.20}$$

With the above choice of sigma points, the first and second moments of the random variable $\boldsymbol{x}$ are preserved through a nonlinear transformation.

In [AH09], the authors present a choice of $2N$ sigma points for the unscented Kalman filter with an intuitive derivation for the sigma points, giving useful insight into the unscented Kalman filter. Their sigma points are the ones in Eq. (2.19) for $\kappa = 0$. However, the very same choice of sigma points had been proposed before in [Sim06].

The unscented Kalman filter outperforms the extended Kalman filter in a wide range of applications [WM00; JU04]. Filters based on the unscented Kalman filter have been applied in various kinds of localization problems, such as [MC05; TBF05]. In [KSC07; KSC08], the authors split the state space in a SLAM problem. The states of landmarks are estimated with unscented Kalman filters, where the measurement model is based on a linearization, though.

The equations of the unscented Kalman filter are presented in Appendix A.2.

2.2.3.3 The Gaussian Sum Filter

Modeling a posterior PDF as a Gaussian PDF is often a poor approximation and degrades the performance of a Bayesian filter. A better approximation is a

weighted sum of $N_{\mathcal{N}}$ Gaussian components, where the ℓ^{th} component has a mean $\boldsymbol{\mu}_{\boldsymbol{x}}^{<\ell>}$, a covariance $\boldsymbol{C}_{\boldsymbol{x}}^{<\ell>}$, and an associated weight $w^{<\ell>}$. The posterior PDF of the state's history given the history of measurements $\boldsymbol{z}_{1:k}$ at time instants one to k is the sum

$$\mathrm{p}\left(\boldsymbol{x}_{0:k}|\boldsymbol{z}_{1:k}\right) = \sum_{\ell=1}^{N_{\mathcal{N}}} w_{0:k}^{<\ell>} \mathcal{N}\Big(\boldsymbol{x}_{0:k}; \boldsymbol{\mu}_{\boldsymbol{x},0:k}^{<\ell>}, \boldsymbol{C}_{\boldsymbol{x},0:k}^{<\ell>}\Big), \tag{2.21}$$

where the weights at each time instant sum up to one. The model in Eq. (2.21) is referred to as Gaussian mixture model [AM12]. The Gaussian sum filter uses Kalman filters to estimate the evolution of the single Gaussian components in the Gaussian mixture. Other Kalman based filters such as the extended Kalman filter or unscented Kalman filter can be used in a straightforward manner. In the update step of the Kalman filters, the weights of the components are updated based on the measurements and normalized to one afterward. The equations for the weight update can be found in [AM12] for the extended Kalman filter case. In [Leo+13] the authors propose a Gaussian sum unscented Kalman filter. The equations for the weight updates in the unscented Kalman filter case can be found in [Kot14], for example.

2.2.3.4 The Particle Filter

The particle filter [GSS93] is a Monte Carlo (MC) based method to solve the recursive Bayesian estimation problem [Gus+02; Aru+02]. The idea of the particle filter is to represent the involved PDFs by a set of random, weighted samples, the so-called particles, in the state space. Each particle can be regarded as a hypothesis of the true state. The history of the state of the i^{th} particle at time instant k is denoted by $\boldsymbol{x}_{0:k}^{<i>}$ and their weights by $w_{0:k}^{<i>}$. The number of particles is N_p. The posterior PDF at time instant k is represented by

$$\mathrm{p}\left(\boldsymbol{x}_{0:k}|\boldsymbol{z}_{1:k}\right) = \sum_{i=1}^{N_p} w_{0:k}^{<i>} \delta\left(\boldsymbol{x}_{0:k} - \boldsymbol{x}_{0:k}^{<i>}\right). \tag{2.22}$$

A PDF represented as in Eq. (2.22) is denoted by the term particle cloud.

With an increasing number of particles, the representation in Eq. (2.22) becomes more accurate, and the non-optimal particle filter approaches the performance of an optimal estimator.

The representation of the involved PDFs in the particle filter allows in principle for a simple implementation of the system and measurement models in Eqs. (2.2) and (2.4). In the prediction step, new particles are drawn based on the system model. In the update step, the weights of the particles are updated based on the measurements. A particle filter can handle both nonlinear models and system and measurement noise terms that are distributed arbitrarily.

It is in general difficult or computationally complex to draw samples from an arbitrary PDF, though. Instead of sampling directly from the PDF on the left hand side of Eq. (2.22), one can sample particles from an importance density $\mathrm{q}\left(\cdot\right)$, and compensate for the mismatch by adjusting the particles' weights.

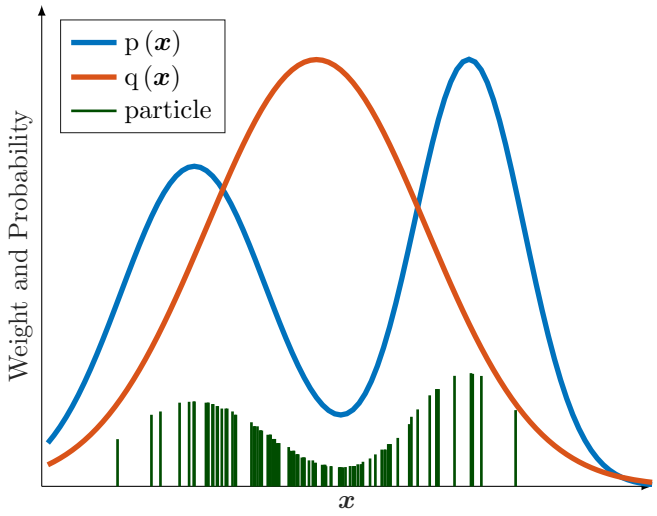


Figure 2.11: If sampling from the PDF $\mathrm{p}(\boldsymbol{x})$ is difficult or complex, particles can be drawn from the importance density $\mathrm{q}(\boldsymbol{x})$. The weights of the particles are adjusted accordingly.

This principle is called importance sampling, and the weights in Eq. (2.22) are calculated by

$$w_k^{<i>} \propto \frac{\mathrm{p}\left(\boldsymbol{x}_{0:k}^{<i>}|\boldsymbol{z}_{1:k}\right)}{\mathrm{q}\left(\boldsymbol{x}_{0:k}^{<i>}|\boldsymbol{z}_{1:k}\right)}. \tag{2.23}$$

Fig. 2.11 illustrates the idea of importance sampling. Instead of directly sampling particles from the PDF $\mathrm{p}(\boldsymbol{x})$, samples are taken from the importance density $\mathrm{q}(\boldsymbol{x})$, and the weights are adjusted according to

$$w^{<i>} \propto \frac{\mathrm{p}\left(\boldsymbol{x}^{<i>}\right)}{\mathrm{q}\left(\boldsymbol{x}^{<i>}\right)}. \tag{2.24}$$

A good importance density is crucial in the design of a particle filter. On the one hand, it should be chosen such that sampling from it is simple. On the other hand, it should be close to proportional to the actual posterior PDF.

The weights in Eq. (2.23) are based on the entire history of user positions. It would be favorable to be able to update the weights sequentially at each time instant k. With the assumption of a first order Markov process and a causal system, the importance density is

$$\mathrm{q}\left(\boldsymbol{x}_{0:k}|\boldsymbol{z}_{1:k}\right) = \mathrm{q}\left(\boldsymbol{x}_k|\boldsymbol{x}_{0:k-1}, \boldsymbol{z}_{1:k}\right) \mathrm{q}\left(\boldsymbol{x}_{0:k-1}|\boldsymbol{z}_{1:k-1}\right). \tag{2.25}$$

In this case, it can be shown [Aru+02] that the weights can be updated as

$$w_k^{<i>} \propto w_{k-1}^{<i>} \frac{\mathrm{p}\left(\boldsymbol{z}_k|\boldsymbol{x}_k^{<i>}\right) \mathrm{p}\left(\boldsymbol{x}_k^{<i>}|\boldsymbol{x}_{k-1}^{<i>}\right)}{\mathrm{q}\left(\boldsymbol{x}_k^{<i>}|\boldsymbol{x}_{0:k-1}^{<i>}, \boldsymbol{z}_{1:k}\right)}. \tag{2.26}$$

In many problems, the importance density is chosen such that

$$\mathrm{q}\left(\boldsymbol{x}_k|\boldsymbol{x}_{0:k-1}^{<i>}, \boldsymbol{z}_{1:k}\right) \coloneqq \mathrm{p}\left(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{<i>}\right), \tag{2.27}$$

where the right hand side can be obtained from the system model. In this case, it becomes simple to draw new particles, and the weight update simplifies to

$$w_k^{<i>} \propto w_{k-1}^{<i>} \, \mathrm{p}\left(\boldsymbol{z}_k|\boldsymbol{x}_{0:k}^{<i>}, \boldsymbol{z}_{1:k-1}\right). \tag{2.28}$$

After the weights are updated, they need to be normalized. The weight of the i^{th} particle is therefore

$$w_k^{<i>} = \frac{w_k^{<i>}}{\sum_{\hat{i}=1}^{N_p} w_k^{<\hat{i}>}}. \tag{2.29}$$

A particle filter typically suffers from degeneracy. Over time, it is likely that the weight of one particle approaches one, while the weights of all other particles go to zero. Hence, the approximation of the posterior PDF becomes very poor, while a large part of the computational complexity is spent on updating particles with negligible contribution. To cope with this problem, one can resample the particles based on their weights to eliminate particles with low weight and thus limited contribution. One possible resampling algorithm is exemplarily illustrated in Fig. 2.12. First, the cumulative distribution function (CDF) sequence $\boldsymbol{c}$ of the particle weights is constructed, such that

$$c_s = \sum_{i=1}^{s} w_k^{<i>} \tag{2.30}$$

for $s = 1, \ldots, N_p$, and $c_0 = 0$. The weight CDF sequence is plotted in blue. The weights $w_i = w^{<i>}$ correspond to the vertical distances between values of the CDF sequence. Likewise, a sequence $\boldsymbol{u}$ of N_p values is chosen such that u_1 is drawn from a uniform distribution between zero and $1/N_p$, and

$$u_i = u_1 + \frac{i-1}{N_p} \tag{2.31}$$

for $i = 2, \ldots, N_p$. The values of $\boldsymbol{u}$ are depicted in red. The first value u_1 is random and determines the other values of the sequence. The differences between neighboring entries in the sequence $\boldsymbol{u}$ are $1/N_p$.

For simplicity, a particle before resampling is denoted by the term *old* particle, and a particle after resampling by the term *new* particle. The i^{th} new particle $\boldsymbol{x}_k^{*<i>}$ is chosen such that it is the $\hat{i}^{\text{th}}$ old particle $\boldsymbol{x}_k^{<\hat{i}>}$ if $c_{\hat{i}-1} < u_i \leq c_{\hat{i}}$. A resampling algorithm is presented in Algorithm 1.

For example, in Fig. 2.12, $c_1 < u_3 \leq c_2$, and thus the third new particle is the second old particle. In particular, some particles might be eliminated or duplicated once or multiple times. In Fig. 2.12, the second old particle is duplicated, and corresponds to the second and third new particle. Therefore, the assignments $\boldsymbol{x}_k^{*<2>} = \boldsymbol{x}_k^{<2>}$ and $\boldsymbol{x}_k^{*<3>} = \boldsymbol{x}_k^{<2>}$ are obtained. The third

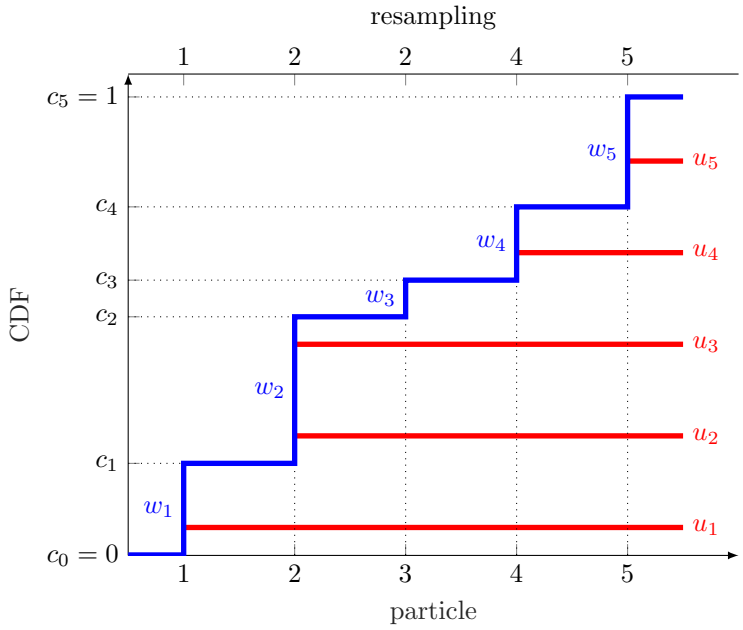


Figure 2.12: The blue curve corresponds to a CDF of the particle weights. The particles are resampled based on the sequence $\boldsymbol{u}$ drawn in red.

Algorithm 1: Resampling algorithm

Input: old particles $\boldsymbol{x}_k^{<i>}$ and weights $w_k^{<i>}$ for $i = 1, ..., N_p$
Output: new particles $\boldsymbol{x}_k^{*<i>}$ and weights $w_k^{*<i>}$ for $i = 1, ..., N_p$

construct the CDF sequence $\boldsymbol{c}$ with $c_0 = 0$ and Eq. (2.30);
set $l = 1$;
draw u_1 from a uniform distribution between zero and $1/N_p$;
construct the sequence $\boldsymbol{u}$ with Eq. (2.31);
for $i = 1, ..., N_p$ **do**
  while $u_i > c_l$ **do**
    l=l+1;
  $\boldsymbol{x}_k^{*<i>} = \boldsymbol{x}_k^{<l>}$;
  $w_k^{*<i>} = 1/N_p$;

old particle $\boldsymbol{x}_k^{<3>}$ is eliminated. The further assignments in the example in Fig. 2.12 are $\boldsymbol{x}_k^{*<1>} = \boldsymbol{x}_k^{<1>}$, $\boldsymbol{x}_k^{*<4>} = \boldsymbol{x}_k^{<4>}$ and $\boldsymbol{x}_k^{*<5>} = \boldsymbol{x}_k^{<5>}$. In the resampling process, the weights of the new particles are set uniformly to $1/N_p$ as in Line 9 in Algorithm 1.

The resampling algorithm is applied if the degeneracy becomes too large. A possible measure for the degeneracy is the effective sample size N_p^{eff} [KLW94], which can be estimated as

$$\hat{N}_p^{\text{eff}} = \frac{1}{\sum_{i=1}^{N_p} \left(w^{<i>}\right)^2}, \tag{2.32}$$

where the weights $w^{<i>}$ are normalized to one. When $\hat{N}_p^{\text{eff}}$ falls below a threshold $\bar{N}_p^{\text{eff}}$, the particles may be resampled [Aru+02]. In the special case where $w_k^{<i>} = 1/N_p$ for $i = 1, \ldots, N_p$, the estimated effective sample size $\hat{N}_p^{\text{eff}}$ equals the actual sample size N_p. This special case occurs after resampling as in Algorithm 1, for example.

One iteration of a generic particle filter at time instant $k > 0$ is summarized in Algorithm 2. There are numerous variants of the particle filter in the literature. They differ by the choice of the importance density, by the time instants for resampling, or the resampling procedure itself, for example.

Algorithm 2: A generic particle filter step at time instant $k > 0$

Input: $\boldsymbol{x}_{0:k-1}^{<i>}$ and $w_{0:k-1}^{<i>}$ for $i = 1, ..., N_p$, $\boldsymbol{z}_{1:k}$
Output: $\boldsymbol{x}_k^{<i>}$ and $w_k^{<i>}$ for $i = 1, ..., N_p$
1 **for** $i = 1, ..., N_p$ **do**
2 | draw new particle $\boldsymbol{x}_k^{<i>}$ from $\mathrm{q}\left(\boldsymbol{x}_k | \boldsymbol{x}_{0:k-1}^{<i>}, \boldsymbol{z}_{1:k}\right)$;
3 | calculate the weight $w_k^{<i>}$ with Eq. (2.28);
4 normalize the weights with Eq. (2.29) ;
5 **if** *the particles suffer from degeneracy* **then**
6 | resample the particles

The particle filter and the unscented Kalman filter seem to bear resemblance to each other as they use points in the state space that are propagated through nonlinear models. However, the idea of the two is fundamentally different. In the unscented Kalman filter, a set of relatively few points is chosen deterministically with the goal of preserving certain moments of the original PDF through the transformation. In the particle filter, a typically large set of points is chosen randomly. Fig. 2.13 illustrates the two different concepts. On the left hand side, a PDF is indicated by its mean depicted by the red cross and the 95 % confidence region depicted by the red ellipse. The probability that a sample taken from the PDF lies within the ellipse is therefore 95 %, and thus the concept of a confidence region is closely related to the covariance matrix. The green dots are 100 samples of this PDF, which can be interpreted as particles. The blue diamond-shaped points are the sigma points from Eq. (2.19). The first sigma point is filled

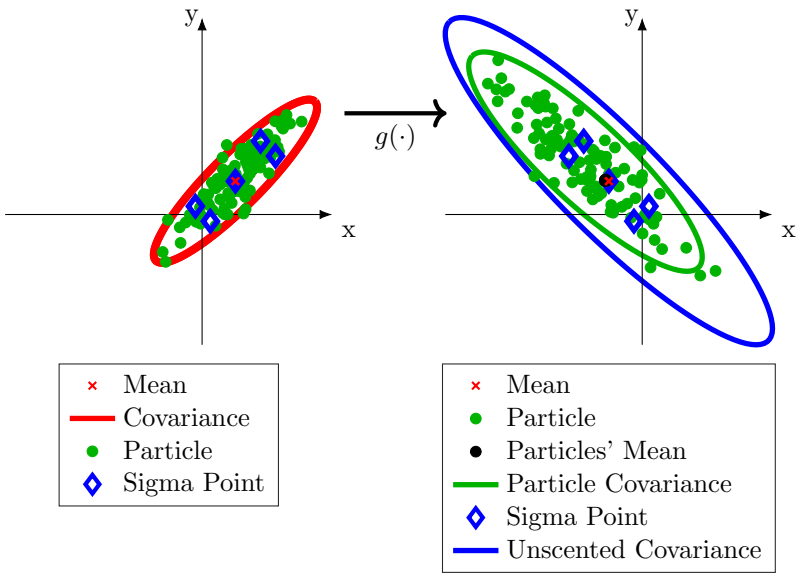


Figure 2.13: The mean and 95 % confidence region of a normal distribution are plotted in red on the left hand side. In addition, 100 particles drawn in green were sampled, and five sigma points, drawn in blue, are plotted. The original mean, the particles and the sigma points are propagated through the nonlinear function $g(\cdot)$ onto the right hand side. The green and blue ellipses are the 95 % confidence regions of the reconstructed covariances based on the particles and sigma points, respectively. The transformed mean, the first sigma point and the particles' mean on the right hand side are in the same location.

blue and coincides with the mean of the PDF. All points and particles on the right hand side have undergone the nonlinear transformation $g(\cdot)$. The mean and the first sigma point still coincide after the transformation. The mean of the transformed particles is almost the same as the transformed mean. The green and blue ellipses represent the 95 % confidence regions reconstructed from the transformed particles and sigma points, respectively. While the mean of the transformed particles and the first transformed sigma points are almost the same, there is some discrepancy in the reconstructed 95 % confidence regions.

Particle filters are applied in a vast range of applications. In [CS13], they are used in structural health monitoring. The authors of [Jou+16] discuss particle filters for prognostics. In biology and chemistry, particle filters are used to track complex cellular processes [She+06] or polymerization processes [Che+04], for example. Further applications include tracking of the human body [MCS02] or human lips [CN10]. In economics, particle filters are used for dynamic stochastic general equilibrium modeling [AT10], to track dynamic macroeconomic models [FR07], or to estimate financial time series such as stock return [SS03], for example.

2.2.4 Simultaneous Localization and Mapping

The problem at hand in SLAM is to localize a user in an unknown environment and to simultaneously create a consistent map of the environment. The user in the SLAM problem was originally a mobile robot, while applications of SLAM include indoor and outdoor, underwater, vehicular or airborne scenarios, for example [DB06]. Likewise, the sensors and types of maps and measurements used for SLAM vary widely.

The map is defined here as the set $\boldsymbol{M}$ of landmarks, or anchors, where the j^{th} landmark is denoted by $\boldsymbol{M}^{<j>}$. Depending on the sensor and system, such a landmark could be a visual feature, a magnetic signature, or a radio transmitter, for example. In SLAM, the goal is to estimate the posterior PDF of the user locations $\boldsymbol{x}_{0:k}$ and the map $\boldsymbol{M}$ based on measurements $\boldsymbol{z}_{1:k}$,

$$\mathrm{p}\left(\boldsymbol{x}_{0:k}, \boldsymbol{M} | \boldsymbol{z}_{1:k}\right). \tag{2.33}$$

The structure of Eq. (2.33) resembles the structure of recursive Bayesian estimation in Eq. (2.11). Indeed, recursive Bayesian estimation can be directly applied to solve the SLAM problem, for example in form of an extended Kalman filter. However, the computational complexity of such an approach grows quadratically with the number of states, making a direct application of recursive Bayesian estimation unfeasible for many applications.

With an algorithm named FastSLAM, the authors of [Mon+02] have introduced a SLAM approach that reduces the state space by Rao-Blackwellization. The key idea is to factorize the SLAM posterior in Eq. (2.33) to

$$\mathrm{p}\left(\boldsymbol{x}_{0:k}, \boldsymbol{M} | \boldsymbol{z}_{1:k}\right) = \mathrm{p}\left(\boldsymbol{M} | \boldsymbol{x}_{0:k}, \boldsymbol{z}_{1:k}\right) \mathrm{p}\left(\boldsymbol{x}_{0:k} | \boldsymbol{z}_{1:k}\right). \tag{2.34}$$

This factorization splits the user state from the map, reducing the dimension of the problem. FastSLAM uses a particle filter to track the state of the user corresponding to the second factor on the right hand side of Eq. (2.34), and extended Kalman filters to estimate the landmarks' locations corresponding to the first factor on the right hand side of Eq. (2.34). Each user particle is not only a hypothesis of the user state, but also carries an estimate of the landmark states, independent from the other particles. The landmark state estimates $\boldsymbol{M}$ are conditioned on the corresponding user particle history $\boldsymbol{x}_{0:k}^{<i>}$. A filter with such a structure is called a Rao-Blackwellized particle filter.

If the measurements for the single landmarks are uncorrelated, the landmarks can be estimated independently from each other. The first factor at the right hand side of Eq. (2.34) can then be written as

$$\mathrm{p}\left(\boldsymbol{M} | \boldsymbol{x}_{0:k}^{<i>}, \boldsymbol{z}_{1:k}\right) = \prod_{j=1}^{N_L} \mathrm{p}\left(\boldsymbol{M}^{<j>} | \boldsymbol{x}_{0:k}^{<i>}, \boldsymbol{z}_{1:k}\right), \tag{2.35}$$

which further decreases the complexity drastically. In this case, though, the spatial correlations among the landmarks can not be used. If an extended Kalman filter is used to estimate the landmarks' state, this means that the

covariance matrix of the map becomes a block diagonal matrix, where each block on the diagonal is the covariance of one single landmark.

The original FastSLAM algorithm uses the movement model of the user as importance density in the particle filter as in Eq. (2.27). The full algorithm with a derivation can be found in [Mon+02]. An updated version with a different importance density can be found in [Mon+03].

The research in SLAM has been driven mainly by the field of robotics, where a robot location is estimated simultaneously with a map of the environment [DB06; BD06]. A variety of different sensors, platforms and applications have been used, and SLAM schemes have been tailored to all kinds of problems.

Autonomous robots doing SLAM are particularly helpful in hostile environments, such as under water [Gut+14], underground [Thr+05], in space [NCR19] or as assistance for humans on rescue missions [BBS10]. Depending on the sensors used, maps can be of various kinds, for example of magnetic field strengths [HSE17] or radio sources [Men+09]. Overviews of developments and future research activities in SLAM are provided in [Cad+16] and [HD16].

2.3 Summary and Outlook

Within this chapter, we have had a look at several aspects of wireless localization and estimation theory. The concept of ToA and AoA positioning will be used throughout the thesis. The performance issues that were mentioned are important to understand the concept of multipath assisted positioning.

All tracking filters used in the following are based on recursive Bayesian estimation. In particular, the particle filter will recur throughout the thesis. In the following chapter, it is used in the multipath assisted positioning scheme named Channel-SLAM, which forms the basis of this thesis. Channel-SLAM is based on SLAM, as it estimates the state of a user simultaneously with creating a map of locations of physical and virtual transmitters.

Chapter 3

Multipath Assisted Positioning and Channel-SLAM

Multipath propagation has so far been considered detrimental for wireless localization. In Section 2.1.3, it was shown how it decreases the positioning performance for signals with finite bandwidths. However, multipath propagation carries spatial information about the environment. The idea of multipath assisted positioning is to exploit this spatial information for localization.

Within this chapter, a wireless radio frequency localization system exploiting multipath propagation is introduced. The network node(s) transmit radio frequency signals and are called transmitters. A user equipped with a receiver utilizes these signals for positioning.

Section 3.1 introduces the idea behind multipath assisted positioning. The multipath signal model is presented in Section 3.2. An overview of the Channel-SLAM algorithm is given in Section 3.3. The two steps of Channel-SLAM, namely channel parameter estimation and state estimation, are explained in detail in Section 3.4 and Section 3.5, respectively. Section 3.6 presents extensions to the Channel-SLAM algorithm.

3.1 The Idea of Multipath Assisted Positioning

The key idea of multipath assisted positioning is to regard MPCs arriving at a receiver as LoS signals from virtual transmitters. Fig. 3.1 illustrates this idea with one physical transmitter labeled Tx, one reflecting wall and one point scatterer. The wall represents a planar surface that reflects signals in a specular manner. The model of a point scatterer is such that it distributes the energy of an impinging signal uniformly to all directions. The user moves along their trajectory and receives the transmitted signal at two different locations. For clarity, a possible LoS signal from the physical transmitter to the user is omit-

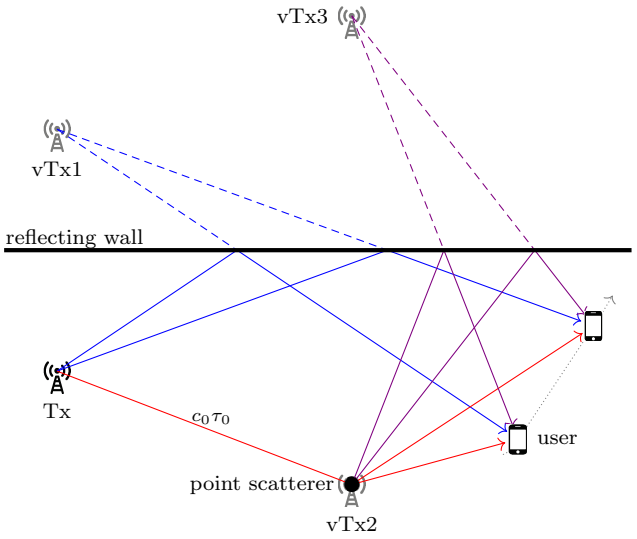


Figure 3.1: The signal from the physical transmitter Tx arrives at the user via three different propagation paths. The first MPC that is reflected at the wall and drawn in blue is regarded as a LoS signal from the virtual transmitter vTx1. On its second propagation path, the signal is scattered at a point scatterer. The user regards the corresponding MPC as a signal from vTx2 in a LoS condition. In the third case, the signal is scattered at the point scatterer and afterwards reflected at the wall before arriving at the user. The MPC is regarded as the LoS component from vTx3.

ted. The signal arrives at the user as a superposition of three different MPCs, which correspond to three different propagation paths that are discussed in the following.

The first propagation path in blue leads from the physical transmitter to the user via a reflection at the reflecting wall. The corresponding signal component is thus a MPC. However, the user treats this signal component as a signal in a LoS condition from the virtual transmitter vTx1. The location of the virtual transmitter is the location of the physical one mirrored at the reflecting wall. Although the reflection point at the wall depends on the position of the user, the location of the virtual transmitter does not. In fact, the location of the virtual transmitter is static if the physical transmitter and the wall are static as well. Furthermore, the two transmitters Tx and vTx1 are perfectly time synchronized. This can be deduced from the observation that the propagation distance of a signal from the physical transmitter to the user via the reflection point at the wall is exactly the same as the LoS propagation distance from the virtual transmitter to the user.

On the second propagation path in red, the transmit signal is scattered at the point scatterer. Again, the corresponding MPC at the user is regarded as a signal from the virtual transmitter vTx2 in a LoS condition. The location of the virtual transmitter vTx2 is the location of the point scatterer, and hence is static as well. Though, there is a propagation delay offset τ_0 between the physical transmitter and the virtual transmitter vTx2. In contrast to the case where the signal is reflected, the propagation distance from the physical transmitter Tx to the user via the scatterer is different from the propagation distance of the signal from the virtual transmitter vTx2 to the user. Their difference is the Euclidean distance d_0 between the two transmitters Tx and Tx2 and can be regarded as an additional propagation distance. The corresponding propagation delay $\tau_0 = {}^{d_0}/_{c_0}$ can be interpreted as a clock offset. In the sense of virtual transmitters, diffraction of the transmit signal can be regarded as scattering.

On the third propagation path in violet, the transmit signal is first scattered at the point scatterer, and afterward reflected at the wall. In this case, the single cases can be regarded one after another as described above. First, the signal is scattered, and the corresponding virtual transmitter is vTx2. Then the signal is reflected, and the resulting virtual transmitter vTx3 is at the location of transmitter vTx2 mirrored at the wall. One can thus observe that the propagation delay offset between the transmitters vTx2 and vTx3 is zero, and the delay offset between the transmitters Tx and vTx3 is τ_0. In this way, the position and propagation delay offset between the physical and a virtual transmitter can be obtained for any combination of specular reflections at surfaces and scattering at point scatterers. If the signal undergoes only reflections along its propagation path, the physical and the resulting virtual transmitter are perfectly time synchronized. If the signal undergoes scattering as well, the length of the actual propagation path from the physical transmitter to the last involved scatterer determines the delay offset between the physical and the corresponding virtual transmitter.

In a nutshell, multipath assisted positioning regards every signal component

arriving at a receiver as a signal in a pure LoS condition from a physical or virtual transmitter. A pure LoS condition between a transmitter and the user's receiver describes the case in which there is only a LoS signal and no MPCs. As a consequence, there are no MPCs in multipath assisted positioning, but the user is either in a pure LoS condition to a transmitter or does not receive a signal from the transmitter at all.

The Channel-SLAM algorithm that is introduced in the following sections does not differentiate between physical and virtual transmitters. The term transmitter by itself therefore refers to either a physical or a virtual transmitter. Likewise, the term signal component refers to either a LoS component or a MPC.

3.2 Signal Model

The physical propagation channel that is considered is a linear, time-variant multipath channel. The transmit signal from a physical transmitter arrives at the receiving antenna array as a superposition of different signal components, each of them traveling on a different propagation path. The ToA of the j^{th} signal component at time instant k is denoted by $\tau_k^{<j>}$, its AoA by $\theta_k^{<j>}$, its complex amplitude by $\alpha_k^{<j>}$, and the array response of the m^{th} antenna element given the AoA θ by $\boldsymbol{a}_m(\theta)$. Neglecting noise terms, the CIR $h_m(\tau, k)$ at time instant k with respect to the m^{th} receiver antenna in the array can then be written as a sum over the signal components,

$$h_m(\tau, k) = \sum_j \alpha_k^{<j>} a_m\left(\theta_k^{<j>}\right) \delta\left(\tau - \tau_{k,m}^{<j>}\right). \tag{3.1}$$

In theory, the number of signal components arriving at the user can be infinite. In real systems, the receiver sensitivity and noise limit the number of detectable signal components. When the transmit signal $s(t)$ is sent over the channel, it is convolved with the CIR. The received signal is sampled at time instants k. The resulting received signal $r_m(\tau, k)$ at time instant k at the m^{th} element is then the sum over the $N_{\text{TX},k}$ detectable signal components

$$r_m(\tau, k) = \sum_{j=1}^{N_{\text{TX},k}} \alpha_k^{<j>} a_m\left(\theta_k^{<j>}\right) s\left(\tau - \tau_{k,m}^{<j>}\right) + n(\tau), \tag{3.2}$$

where $n(\tau)$ is additive noise, which includes both additive white Gaussian noise and dense multipath components [Ric05]. In Eq. (3.2), it is assumed that the channel is static during the short time interval during which a snapshot of the signal is sampled at time instant k.

3.3 Overview of Channel-SLAM

The authors of [GJD13a; Gen+16a] have introduced Channel-SLAM as a multipath assisted positioning algorithm to track a single user in a multipath scenario. Channel-SLAM considers static physical and virtual transmitters. The

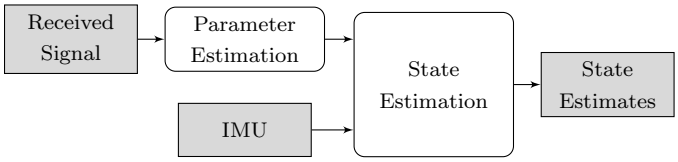


Figure 3.2: Channel-SLAM is a two step approach. In the fist step, the parameters of the arriving signal components are estimated based on the received signal. In the second step, these estimates are used to jointly estimate the user state and the transmitter states with SLAM. In addition, further sensors such as from an inertial measurement unit may be used.

user in the scenario is equipped with a radio frequency receiver that is time synchronized to the physical transmitter. In the case that there is no time synchronization between the receiver and the physical transmitter, the time offset and possibly its derivatives are estimated as well. Within this thesis, such a time synchronization is assumed. In the following, scenarios with only one physical transmitter are considered.

Channel-SLAM does not differentiate between the LoS component and MPCs of a physical transmitter. Each signal component arriving at the user corresponds to one transmitter. Hence, there is no differentiation between physical and virtual transmitters, and all transmitters are covered by the same model. Each transmitter is characterized by its location and delay offset as explained in Section 3.1.

In Channel-SLAM, no knowledge about the location of the physical transmitter or a floor plan to calculate transmitter locations is required. Thus, Channel-SLAM estimates both the user position and velocity in two dimensions jointly with the transmitter locations and clock offsets.

An overview of the Channel-SLAM algorithm is depicted in Fig. 3.2. Channel-SLAM is a two step approach. In a first step, the parameters of signal components are estimated by a channel estimator. Details on the channel parameter estimation are provided in Section 3.4. The estimated ToAs and, if an antenna array is available at the receiver, AoAs of the signal components are then used as measurements in the second step, where the position and velocity of the user and the locations and delay offsets of the transmitters are estimated jointly with SLAM by means of a Rao-Blackwellized particle filter. Additional sensors, such as an inertial measurement unit (IMU), can be used in this second step, which is described in detail in Section 3.5.

The system of equations considered by Channel-SLAM is underdetermined when one single time instant is considered. Each of the N_{TX} transmitters has three unknowns, namely its two-dimensional location and its clock offset. In a general D-dimensional scenario, the number of unknowns for each transmitter is $D+1$. In addition, the location of the user corresponds to D more unknowns, if the user is time synchronized to the physical transmitter. For the sake of simplicity, estimating the user velocity is omitted in the following consideration. The number of linearly independent equations relating the measured quantities

to the unknowns at one time instant and for one transmitter in the second stage of Channel-SLAM is denoted by E. If only ToA measurements are available from the channel estimator, for example, $E = 1$, and if in addition AoA measurements can be obtained, $E = 2$. To be able to solve the system conceptually, the number of equations EN_{TX} must be at least the number of unknowns, yielding

$$EN_{\mathrm{TX}} \geq \underbrace{(D+1)N_{\mathrm{TX}}}_{\text{unknowns from transmitters}} + \underbrace{D}_{\text{unknown from the user}} . \tag{3.3}$$

A necessary condition for Eq. (3.3) to hold in a $D = 2$ dimensional scenario is $E > 3$. Hence, such a system can not be solved with only $E = 2$ linearly independent measurements. However, as the user moves through the scenario, signals at different user positions are received, and additional linearly independent equations are obtained. At every new time instant, there are only D more unknowns from the user position, but EN_{TX} more equations. With measurements from k different user positions,

$$kEN_{\mathrm{TX}} \geq (D+1)N_{\mathrm{TX}} + kD. \tag{3.4}$$

Rearranging yields the necessary condition $EN_{\mathrm{TX}} > D$, which can be fulfilled with a sufficient number of transmitters, and the entire system of equations can in principle be solved over time.

3.4 Channel Parameter Estimation

In Channel-SLAM, a stationary scenario is assumed. However, the channel is time-variant due to the movement of the receiver. To use the MPCs for positioning, it is important to know how the signal components' parameters evolve over time. In particular, associations among signal components across time instants are required to be able to correctly assign signal components to transmitters. Snapshot-based estimators for estimating parameters of signal components are therefore not well suited.

The channel estimator used within this thesis is the Kalman Enhanced Super Resolution Tracking (KEST) algorithm [Jos+12]. KEST works in two stages. In the inner stage, a Kalman filter tracks the parameters of signal components. In the update stage of the Kalman filter, a snapshot based estimator is used to update the parameter estimates. While KEST can use any ML estimator in the inner stage, the SAGE algorithm is used within this thesis. The outer stage of KEST runs several of these Kalman filters in parallel. Each Kalman filter carries a different hypothesis of the number of signal components in the received signal. Initially, this number of signal components needs to be estimated, which is a model order estimation problem. Afterward, the outer stage of KEST tracks this number over time.

The PDF, or likelihood function, of a snapshot of the received signal r parametrized by $\boldsymbol{\Theta}$ is denoted by $\mathrm{p}(r; \boldsymbol{\Theta})$. The vector $\boldsymbol{\Theta}$ contains parameters that are observable from the received signal, such as ToA, AoA or amplitude, and their derivatives, which are typically non-observable.

3.4.1 Model Order Estimation

The parametric model of a signal is often known, but not the number of parameters in the model. Estimating the number of signal components within a noisy signal is an example for such a model order estimation problem [RR97]. Model order estimation problems have been studied in the literature for decades. The most prominent approach uses information theoretic criteria [SS04]. It is based on a best fit of the received signal and the model, where a criterion for a good fit is based on the Kullback–Leibler divergence, for example. Methods using information theoretic criteria have in common that they try to minimize the term

$$-2 \log \mathrm{p}\left(r; \boldsymbol{\Theta}\right) + \eta\left(L, N_r\right) L, \tag{3.5}$$

where $\eta\left(L, N_r\right)$ is a function of the model order L and the length N_r of the received signal.

Most information theoretic criteria based approaches to model order estimation differ only in the choice of $\eta\left(\cdot\right)$, which can be regarded as a penalty for an increasing model order L. Without such a penalty term, an estimator tends to fit the model to the noise.

Famous selection rules include the Akaike information criterion (AIC), for which $\eta\left(L, N_r\right) = 2$, and the Bayesian information criterion (BIC), for which $\eta\left(L, N_r\right) = \log N_r$ [SS04]. While the AIC might be inconsistent and tends to overestimate L, the BIC is considered a better choice as it is an approximation to the maximum a posteriori (MAP) rule. Within this thesis, the BIC selection rule is used.

3.4.2 Snapshot-Based Channel Parameter Estimation

In the update step of the Kalman filters in the inner stage of KEST, a snapshot based ML estimator is applied to estimate the parameters of signal components. Such estimators are often used in channel sounding [Wan14], and they can be split into three major categories [KV96]. The first category is based on spectral estimation, with the Multiple Signal Classification (MUSIC) [Sch86] algorithm as the most famous representative. The second category are parametric subspace based estimators such as the ESPRIT algorithm. The idea is to separate the signal and noise subspace based on an eigendecomposition. The third category contains deterministic parametric estimators, of which the Expectation-Maximization (EM) algorithm [DLR77] and SAGE [Fle+96] as an extension of EM algorithm are the most prominent examples. Estimators in this category are ML based methods. While ESPRIT and SAGE have been shown to perform very similarly in channel sounding [Tsc+99], we focus on SAGE in the following.

The EM algorithm has been used to estimate a single parameter of signal components [FW88], while the SAGE algorithm can estimate multiple parameters jointly [Fle+96]. The optimum ML estimator can in general not be obtained analytically, or the computational complexity is too high, since too many parameters are involved. The underlying idea of both algorithms is to approximate the

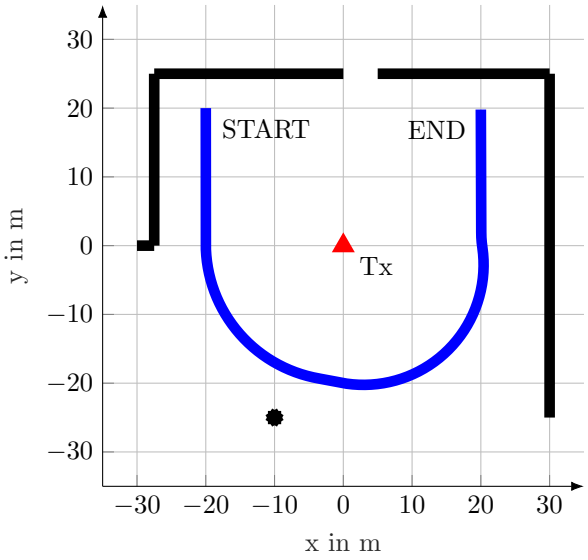


Figure 3.3: In the scenario, there is one physical transmitter labeled Tx. The black lines are walls that reflect the transmit signal, and the black dot represents a point scatterer that distributes the energy of an arriving signal uniformely to all directions. The blue curve from the point labeled START to the point labeled END is a user trajectory.

ML solution iteratively in two steps, namely expectation and maximization, and thus to decompose the estimation problem to problems of smaller dimension. The two steps are briefly described in the following.

In the maximization step, the parameters of each single signal component are obtained from an ML estimator. If only one parameter is estimated, the ML estimator can be obtained in closed form [Fle+99]. While the parameters for one signal component are estimated, the respective other signal components' parameters are assumed fixed and known.

The complete data describes the sum of the signal components forming the noise-free received signal. This complete data is unobservable. Observations of the incomplete data, which is the noisy received signal, can be made. The complete data is then reconstructed from these observations and the previous estimates of the signal components in the expectation step. Therefore, the contribution of the estimated signal components are subtracted from the observed incomplete data. The expectation and maximization step are performed iteratively until convergence is achieved.

In the SAGE algorithm, refining the parameters of each signal component in the maximization step is performed with the EM algorithm.

The SAGE algorithm needs prior knowledge on the number of signal components, which can be obtained from model order estimation schemes in Section 3.4.1, for example.

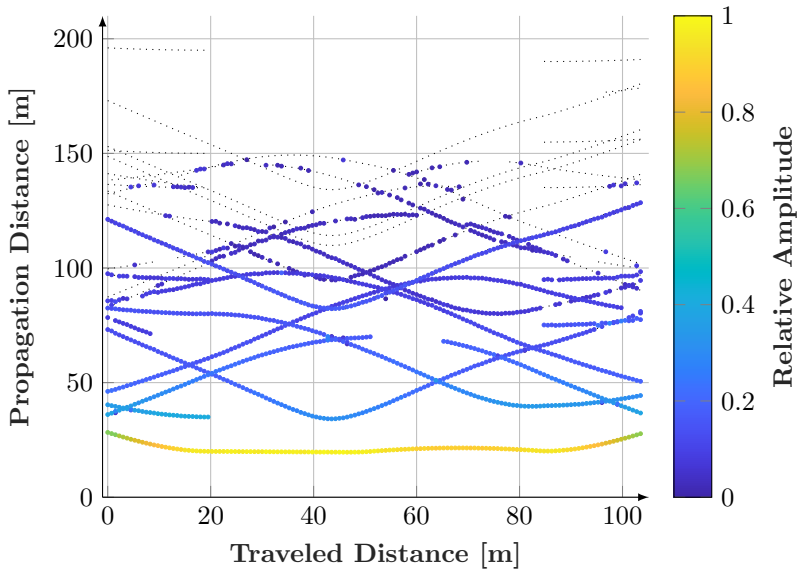


Figure 3.4: The SAGE results are plotted in terms of propagation distance of signal components versus the user traveled distance. The dotted lines indicate the true propagation distances for all propagation paths up to an order of two.

Fig. 3.3 shows an example scenario with one physical transmitter illustrated by the red triangle labeled Tx. The black lines represent walls that reflect the transmit signal, and the black circle represents a point scatterer. A user trajectory is plotted in blue with the start point labeled START and the end point labeled END.

The results of the SAGE algorithm for simulations of a user track in the scenario in Fig. 3.3 are plotted in Fig. 3.4. The user is equipped with a receiver with a rectangular antenna array consisting of nine elements. Every 100 ms, the user records a snapshot of the received signal. The transmitter continuously transmits a signal with a bandwidth of 100 MHz and a uniform power spectrum density. In the simulations, the CIR is calculated for each user location with ray tracing. To create a multipath scenario, single and double reflections and/or scattering are incorporated as the transmit signal is reflected at the walls or scattered at the point scatterer. The SNR of the received power averaged over the user track is 10.1 dB.

In Fig. 3.4, the propagation distances of signal components are plotted versus the distance the user has traveled assuming eight signal components at each time instant. The propagation distance is the ToA of the signal component multiplied by the speed of light. The colors of the curves indicate the absolute values of the amplitudes of the respective signal component relative to the maximum amplitude of all signal components throughout the track in linear domain. At each user location corresponding to a different traveled distance, the propagation

distances of signal components with parameters estimated by SAGE can be obtained. The dotted lines correspond to the true propagation paths for the signal components up to an order of two, i.e., they represent single and double reflections and/or scattering.

The curve with the shortest propagation distance throughout the track corresponds to the LoS path. While the signal components with small propagation distances are estimated with high accuracy, some MPCs, in particular with higher propagation distances, are estimated with less accuracy, only for a certain distance, pointwise, or not at all.

If two or more signal components are closely spaced in several parameters such as ToA, AoA and phase, the corresponding parameter estimates can be biased, or the signal components can not be resolved. In this respect, a high bandwidth of the transmit signal is preferable, since it allows for a better resolution of the signal components in time domain as shown in Section 2.1.3.

3.4.3 Tracking the Model Order

In KEST, the parameters of signal components and their first order derivatives are tracked by a number of parallel Kalman filters to obtain the evolution of such parameters over time. Each of the Kalman filters carries a hypothesis $\hat{L}$ of the number of signal components, i.e., the model order.

In the inner stage of KEST, a Kalman filter corresponding to such a model order hypothesis tracks the mean and covariance of the signal components' parameters. The system model in the Kalman filters is linear and the process noise assumed to be additive white Gaussian noise (AWGN) [Jos14]. In the prediction step, the mean $\hat{\boldsymbol{\Theta}}_{\hat{L},k-1}$ and covariance matrix $\hat{\boldsymbol{P}}_{\hat{L},k-1}$ estimated at the previous time instant undergo the movement model of the Kalman filter, resulting in the predicted mean $\hat{\boldsymbol{\Theta}}^{-}_{\hat{L},k}$ and covariance matrix $\hat{\boldsymbol{P}}^{-}_{\hat{L},k}$. In the update step, these predicted values are fed as initialization into the SAGE algorithm, which returns the corresponding posterior estimates $\hat{\boldsymbol{\Theta}}_{\hat{L},k}$ and $\hat{\boldsymbol{P}}_{\hat{L},k}$ based on the current snapshot of the signal. The detailed equations of the Kalman filters can be found in [Jos+12]. An example of the prediction and update in one Kalman filter with $\hat{L} = 4$ regarding the mean of the ToA of one signal component is illustrated in Fig. 3.5. The figure shows the absolute value of a received signal drawn in blue at time instant $k-1$, and below at time instant k. The amplitude estimates of the four signal components are drawn at the corresponding delays as red solid lines. The estimated mean $\hat{\tau}^{<1>}_{k-1}$ of the delay of the first signal component from time instant $k-1$ undergoes the prediction in the Kalman filter resulting in $\hat{\tau}^{<1>-}_{k}$ indicated by the red dashed line. The gray arrow labeled Prediction visualizes this transition. In the update step of the Kalman filter, the SAGE algorithm is used to obtain the posterior mean $\hat{\tau}^{<1>}_{k}$ as indicated by the gray arrow labeled Update. The predicted values for the delays and amplitudes of the other three signal components at time instant k are drawn as red dotted lines. The Kalman filters track not only the means, but also the covariance matrices of the signal components' parameters, which are not plotted in Fig. 3.5.

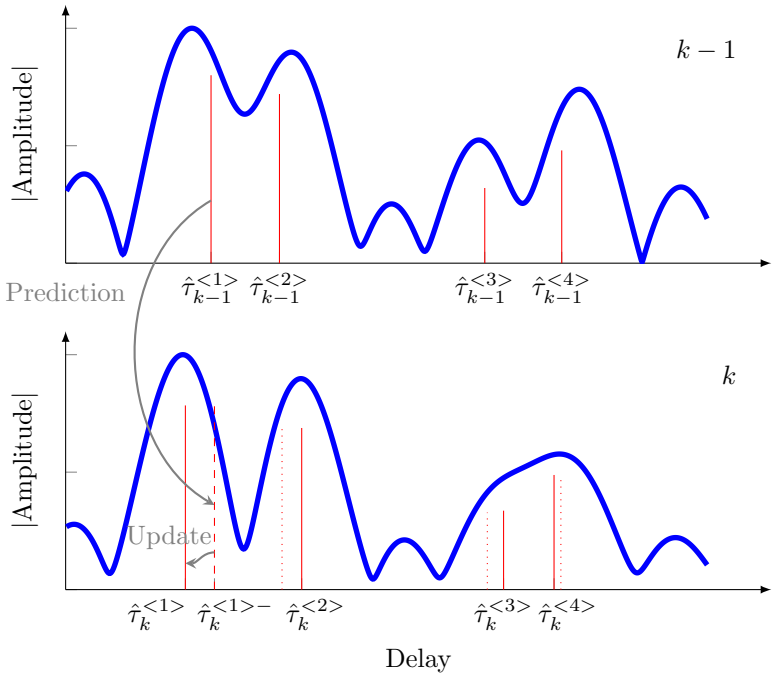


Figure 3.5: The absolute value of a received signal is plotted in blue at time instant $k-1$ and at time instant k to visualize the prediction and update step of KEST. The KEST delay estimates for the four delays $\hat{\tau}$ are plotted for both time instances in red with solid lines. The predicted delay estimate $\hat{\tau}_k^{<1>-}$ at time instant k is indicated by a red dashed line.

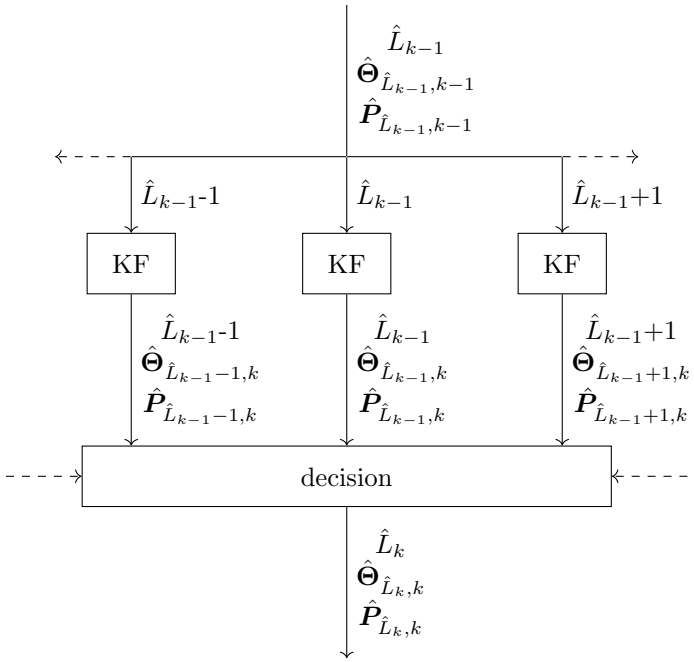


Figure 3.6: In KEST, the signal components' parameters are estimated with Kalman Filters (KFs) in parallel for different numbers of signal components.

Three separate Kalman filter instances as explained above run in parallel with different model orders. The estimated model order at time instant $k-1$ is denoted by $\hat{L}_{k-1}$, and the set of model order hypotheses at time instant k by $\mathcal{L}_k$. At time instant k, the three model order hypotheses in $\mathcal{L}_k$ are $\hat{L}_{k-1}-1$, $\hat{L}_{k-1}$ and $\hat{L}_{k-1}+1$. After one iteration of the Kalman filters, a decision block decides for one of the hypotheses, and thus for the current model order $\hat{L}_k$.

Whenever the model order is chosen as the maximum in $\mathcal{L}_k$ in the decision block, the model order $\max(\mathcal{L}_k)+1$ is added to $\mathcal{L}_k$, and a Kalman filter instance is run with the corresponding highest model order in the new set $\mathcal{L}_k$. Likewise, every time the minimum of $\mathcal{L}_k$ is chosen to be the new model order, the model order $\min(\mathcal{L}_k)-1$ is added to $\mathcal{L}_k$, and a Kalman filter instance is run with the corresponding lowest model order in the new set $\mathcal{L}_k$. Afterward, a new decision on the model order is made based on the new set $\mathcal{L}_k$. Hence, the model order may change from one time instant to another by any number. A flow chart of KEST is depicted in Fig. 3.6. The quantities $\hat{\boldsymbol{\Theta}}_{\hat{L}_k,k}$ and $\hat{\boldsymbol{P}}_{\hat{L}_k,k}$ are the estimated mean and covariance, respectively, of the signal components' parameters for a model order of $\hat{L}_k$.

In the decision block, the residuals between the reconstructed signal for the different model orders and corresponding parameters and the received signal are calculated. If the decision is made for the order and parameters of the lowest residual energy, KEST tends to over-fit the signal, though. Such over-

fit means that noise samples are tracked. In addition, frequent changes of the model order would cause tracking of signal components with low power and a short lifetime [Jos14]. Hence, two penalty terms are introduced. The first one penalizes changes in the model order, and the second one high model orders. The second penalty term is in line with the information criterion for model order selection. The final decision is made for the model order for which the sum of the two penalty terms and an approximation quality term, which is based on the residual energy, is the lowest [Jos+12]. The model order, the corresponding estimates of the parameters and their covariance matrices are used in the next time instant as prior information for the Kalman filter instances.

The change in the model order from time instant $k-1$ to time instant k is denoted by $\delta_{L,k} = \hat{L}_k - \hat{L}_{k-1}$. If the model order does not change, $\delta_{L,k} = 0$.

If $\delta_{L,k} > 0$, additional signal components were detected. The reconstructed signal based on the estimated parameters is subtracted from the received signal. The parameters of the $\delta_{L,k}$ new signal components are sequentially initialized based on the resulting residual. The state vector $\boldsymbol{\Theta}$ is extended accordingly.

If $\delta_{L,k} < 0$, signal components are discarded. The signal components to be discarded are chosen such that the residual when subtracting the reconstructed signal with the remaining signal components from the received signal is the lowest. The corresponding signal components are deleted from the state vector. An example where one signal component is discarded from one time instant to another is illustrated in Fig. 3.7.

The KEST results in terms of propagation distance versus traveled distance for the scenario in Fig. 3.3 with the same signal parameters and user track are plotted in Fig. 3.8. The curve with the shortest propagation distance throughout the track corresponds to the LoS path. In contrast to Fig. 3.4 where SAGE is used for estimating the signal components' parameters, one can observe that in particular signal components with larger propagation distances are tracked more smoothly. Furthermore, the number of signal components is inherently tracked by KEST.

Tracking the parameters of signal components with Kalman filters inherently yields associations of signal components from one time instant to another. In contrast to snapshot-based estimators such as the pure SAGE algorithm, no additional data association schemes that establish relations among signal components for neighboring time instants need to be applied. However, KEST cannot associate a signal component once track of the component is lost. If track of a component is lost, and the signal component corresponding to the same propagation path is detected again at a later time instant, KEST can not establish the corresponding association. Instead, Channel-SLAM initializes a new transmitter for this signal component. The problem of associating signal components to transmitters is referred to as data association, and will be addressed later in the thesis.

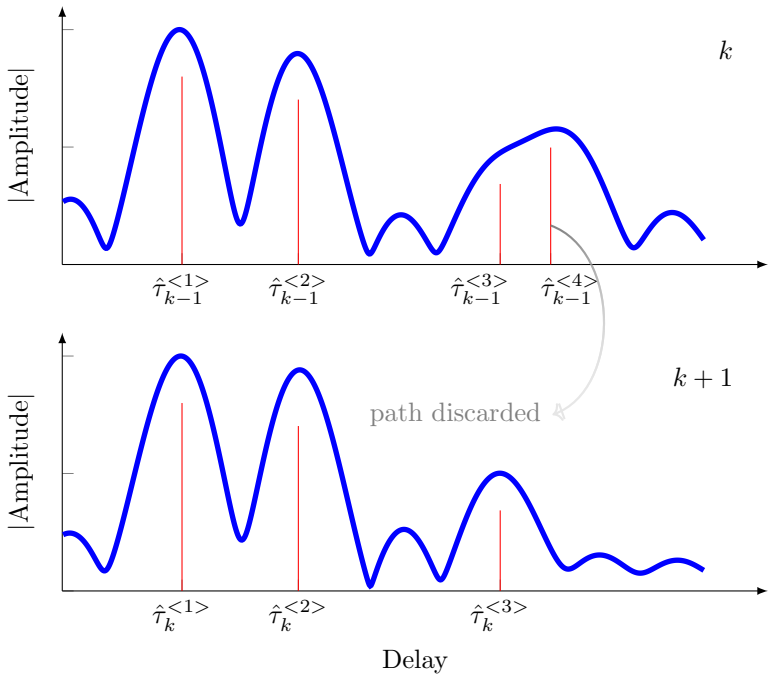


Figure 3.7: The blue curves are the absolute values of received signals at time instants k and $k+1$. The red lines represent estimates of signal components' parameters from KEST. The fourth signal component at time instant k is discarded at time instant $k+1$.

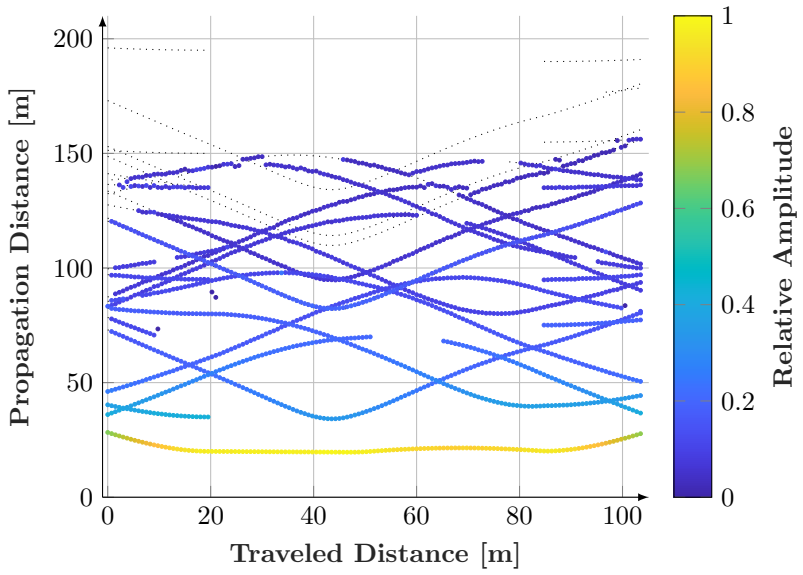


Figure 3.8: The KEST results are plotted in terms of propagation distance of signal components versus the user traveled distance. The thin dotted lines are true propagation distances for all propagation paths up to an order of two.

3.5 State Estimation

The estimates from KEST are used in the second step of Channel-SLAM as measurement input to simultaneously track the user and estimate the states of the transmitters. The number of transmitters that are detected and tracked by KEST at time instant k is denoted by $N_{\mathrm{TX},k}$, and corresponds to the model order estimate $\hat{L}_k$. The overall number of transmitters that have been detected and tracked by KEST from time instants one to k is denoted by $N^{\mathrm{a}}_{\mathrm{TX},k}$. If KEST detects new transmitters at time instant k, $N^{\mathrm{a}}_{\mathrm{TX},k} = N^{\mathrm{a}}_{\mathrm{TX},k-1} + \delta_{L,k}$. Otherwise, $N^{\mathrm{a}}_{\mathrm{TX},k} = N^{\mathrm{a}}_{\mathrm{TX},k-1}$.

3.5.1 Bayesian Formulation

Within this thesis, ToA and AoA measurements are considered in Channel-SLAM. The ToA estimates from KEST at time instant k are stacked in the vector

$$\boldsymbol{\tau}_k = \left[\tau_k^{<1>} \ \ldots \ \tau_k^{<N_{\mathrm{TX},k}>}\right]^T. \tag{3.6}$$

The vector $\boldsymbol{\theta}_k$ contains the corresponding AoA estimates,

$$\boldsymbol{\theta}_k = \left[\theta_k^{<1>} \ \ldots \ \theta_k^{<N_{\mathrm{TX},k}>}\right]^T. \tag{3.7}$$

The final measurement vector obtained from KEST at time instant k is thus

$$\boldsymbol{z}_{\mathrm{R},k} = \left[\boldsymbol{\tau}_k^T \;\; \boldsymbol{\theta}_k^T\right]^T, \tag{3.8}$$

The length of $\boldsymbol{z}_{\mathrm{R},k}$ is $N_{\mathrm{TX},k} N_{\mathrm{m}}$, where N_{m} denotes the number of different types of measurements that are available. For example, if ToA and AoA measurements are used, $N_{\mathrm{m}} = 2$.

The ToA estimates in Eq. (3.6) are expressed here in time domain. They can also be stated in a distance domain as propagation delay estimates

$$\boldsymbol{d}_k = c_0 \left[\tau_k^{<1>} \;\; \ldots \;\; \tau_k^{<N_{\mathrm{TX},k}>}\right]^T = \left[d_k^{<1>} \;\; \ldots \;\; d_k^{<N_{\mathrm{TX},k}>}\right]^T. \tag{3.9}$$

In the following, the ToA estimates' representation in the distance domain are used wherever it provides notational simplicity or a more intuitive understanding.

Channel-SLAM considers a static two-dimensional environment with one physical transmitter. The generalization to multiple physical transmitters is straightforward if the transmit signals are separated in time, frequency or code domain, for example. All transmitters have a static location and clock offset. The location of the j^{th} transmitter at time instant k is denoted in Cartesian coordinates by $\boldsymbol{p}_{\mathrm{TX},k}^{<j>} = \left[x_{\mathrm{TX},k}^{<j>} \;\; y_{\mathrm{TX},k}^{<j>}\right]^T$, and its clock offset by $\tau_{0,k}^{<j>}$. The state vector of the j^{th} transmitter at time instant k is thus

$$\boldsymbol{x}_{\mathrm{TX},k}^{<j>} = \left[x_{\mathrm{TX},k}^{<j>} \;\; y_{\mathrm{TX},k}^{<j>} \;\; \tau_{0,k}^{<j>}\right]^T = \left[\boldsymbol{p}_{\mathrm{TX},k}^{<j>\,T} \;\; \tau_{0,k}^{<j>}\right]^T. \tag{3.10}$$

In Channel-SLAM, both the user location $\boldsymbol{p}_{\mathrm{u},k} = [x_k \;\; y_k]^T$ and the user velocity $\boldsymbol{v}_{\mathrm{u},k} = [v_{x,k} \;\; v_{y,k}]^T$ are tracked. The user state vector at time instant k is

$$\boldsymbol{x}_{\mathrm{u},k} = \left[\boldsymbol{p}_{\mathrm{u},k}^T \;\; \boldsymbol{v}_{\mathrm{u},k}^T\right]^T = [x_k \;\; y_k \;\; v_{x,k} \;\; v_{y,k}]^T. \tag{3.11}$$

The entire state vector consists of the user state and the state of the $N_{\mathrm{TX},k}$ transmitters,

$$\begin{aligned} \boldsymbol{x}_k &= \left[\boldsymbol{x}_{\mathrm{u},k}{}^T \;\; \boldsymbol{x}_{\mathrm{TX},k}^{<1>\,T} \;\; \ldots \;\; \boldsymbol{x}_{\mathrm{TX},k}^{<N_{\mathrm{TX},k}>\,T}\right]^T \\ &= \left[\boldsymbol{x}_{\mathrm{u},k}{}^T \;\; \boldsymbol{x}_{\mathrm{TX},k}{}^T\right]^T. \end{aligned} \tag{3.12}$$

Within this thesis, the user is assumed to be time synchronized to the physical transmitter. It is straightforward to drop this assumption, though. In such a case, the clock offset between the physical transmitter and the user needs to be estimated and to be included in the user state vector [Gen18]. Depending on the model of the clock, not only the clock offset but also its derivatives need to be estimated.

In Channel-SLAM, the history of the state of the user and the states of the transmitters are estimated based on the measurements and, if available, control inputs, which are obtained from other sensors such as an inertial measurement unit. In particular, recursive Bayesian estimation is used to track the PDF

$\mathrm{p}\left(\boldsymbol{x}_{0:k}|\boldsymbol{z}_{\mathrm{R},1:k},\boldsymbol{u}_{1:k}\right)$, where the measurements and the control inputs from time instants one to k are denoted by $\boldsymbol{z}_{\mathrm{R},1:k}$ and $\boldsymbol{u}_{1:k}$, respectively. The vector $\boldsymbol{x}_{0:k}$ denotes the history of the user and transmitter states from time instants zero to k. Regarding the conditional probability, the PDF can be written as the product

$$\begin{aligned}\mathrm{p}\left(\boldsymbol{x}_{0:k}|\boldsymbol{z}_{\mathrm{R},1:k},\boldsymbol{u}_{1:k}\right) &= \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k},\boldsymbol{x}_{\mathrm{u},0:k}|\boldsymbol{z}_{\mathrm{R},1:k},\boldsymbol{u}_{1:k}\right)\\ &= \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}|\boldsymbol{z}_{\mathrm{R},1:k},\boldsymbol{u}_{1:k}\right)\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}|\boldsymbol{x}_{\mathrm{u},0:k},\boldsymbol{z}_{\mathrm{R},1:k}\right)\end{aligned} \tag{3.13}$$

to separate the user space from the transmitter space. The user states can therefore be estimated separately from the transmitters, while the transmitters' states are estimated conditioned on the user states. The control input is assumed independent from the transmitters' states. Factorizing a PDF in this way is called Rao-Blackwellization. As the computational complexity of SLAM typically grows polynomially in the dimension of the state space, this Rao-Blackwellization decreases the computational complexity considerably.

Furthermore, Channel-SLAM assumes independence among the measurements for the single transmitters. Correlations among estimated signal components in KEST occur only if two or more signal components are spaced closely in all signal parameters tracked by KEST, for example if two signal components arrive at the receiver from the same direction with similar ToAs and complex amplitudes. It can be assumed that such situations occur only for short time spans. Thus, correlations affect the estimation of the signal components' parameters only for short time spans as well. Consequently, the KEST estimates and therefore the measurements for the signal components' parameters can be assumed uncorrelated in the long term.

The second factor in the second line of Eq. (3.13), which is the posterior PDF of the transmitters' state, is further decomposed as [Thr+04]

$$\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}|\boldsymbol{x}_{\mathrm{u},0:k},\boldsymbol{z}_{\mathrm{R},1:k}\right) = \prod_{j=1}^{N_{\mathrm{TX},k}} \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<j>}|\boldsymbol{x}_{\mathrm{u},0:k},\boldsymbol{z}_{\mathrm{R},1:k}\right). \tag{3.14}$$

The above factorization decreases the computational complexity for the estimation of the transmitter states drastically. Instead of one estimator estimating $3N_{\mathrm{TX},k}$ quantities, there are now $N_{\mathrm{TX},k}$ estimators for three quantities each.

The MMSE estimator $\hat{\boldsymbol{x}}_{0:k}$ of the state vector $\boldsymbol{x}_{0:k}$ is defined as in Eq. (2.10),

$$\hat{\boldsymbol{x}}_{0:k,\mathrm{MMSE}} = \int \boldsymbol{x}_{0:k}\,\mathrm{p}\left(\boldsymbol{x}_{0:k}|\boldsymbol{z}_{\mathrm{R},1:k},\boldsymbol{u}_{1:k}\right)\mathrm{d}\boldsymbol{x}_{0:k}. \tag{3.15}$$

3.5.2 Rao-Blackwellized Particle Filter

Channel-SLAM uses a Rao-Blackwellized particle filter to recursively estimate the posterior PDF in Eq. (3.13). The particle filter estimating the user posterior PDF is referred to as user particle filter, and its particles as user particles. The user posterior PDF is modeled by

$$\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}|\boldsymbol{z}_{\mathrm{R},1:k},\boldsymbol{u}_{1:k}\right) = \sum_{i=1}^{N_p} w_{0:k}^{<i>}\,\delta\left(\boldsymbol{x}_{\mathrm{u},0:k} - \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}\right), \tag{3.16}$$

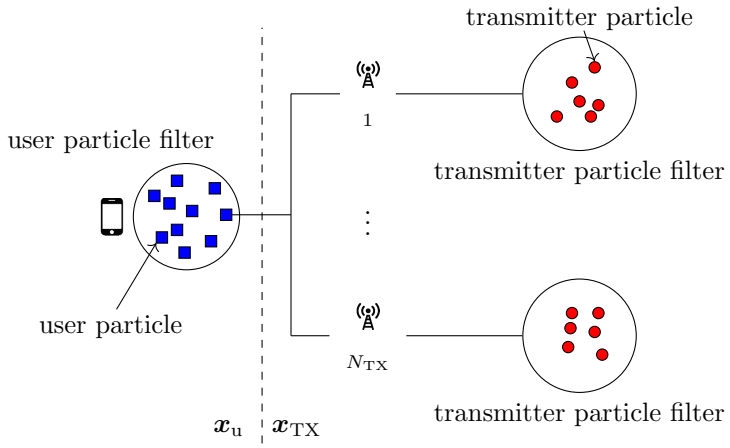


Figure 3.9: The structure of the Rao-Blackwellized particle filter in Channel-SLAM. The states of the transmitters are estimated with particle filters for every user particle in the user particle filter.

where $\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}$ is the i^{th} user particle history and $w_{0:k}^{<i>}$ its associated weight. The number of particles in the user particle filter is denoted by N_p. The information regarding the i^{th} user particle comprises the particle itself and the transmitter states for that particle,

$$\left\langle \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \left\{ \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{R},1:k} \right) \right\}_{j=1,\ldots,N_{\mathrm{TX},k}^{\mathrm{a}}} \right\rangle, \tag{3.17}$$

where $\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>}$ is the history of states of the j^{th} transmitter regarding the i^{th} user particle. Hence, the user state history for the i^{th} particle is assumed to be given by the user particle history $\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}$, and the transmitter states are estimated assuming that $\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}$ is the true user state history.

The structure of the Rao-Blackwellized particle filter is depicted in Fig. 3.9. As indicated by Eq. (3.13), the user state space on the left is separated from the transmitter state space on the right. The transmitters' states are estimated for each user particle independently from the other user particles. In Fig. 3.9, the structure of the estimation of the transmitter states with N_{TX} transmitters is depicted for one user particle. Since the measurements for the transmitters are assumed independent as in Eq. (3.14), each transmitter state is estimated independently from the other transmitter states. The state of each transmitter is estimated by a particle filter. Therefore, the posterior PDF of the j^{th} transmitter given the state vector of the i^{th} user particle is modeled by

$$\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{R},1:k} \right) = \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{0:k}^{<i,j,\ell>} \delta\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>} - \boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j,\ell>} \right), \tag{3.18}$$

where $\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j,\ell>}$ is the history of the ℓ^{th} particle in the particle filter for the j^{th} transmitter and the i^{th} user particle, and $w_{0:k}^{<i,j,\ell>}$ its associated weight.

A particle filter estimating the state of a transmitter is denoted by the term transmitter particle filter, and a corresponding particle by transmitter particle. The number $N_{p,\mathrm{Tx}}$ of transmitter particles may differ for different time instants, different user particles and different transmitters. For notational simplicity, the corresponding indices are omitted in $N_{p,\mathrm{Tx}}$.

The transmitters are assumed to be static in Channel-SLAM. For numerical stability though, a very slow movement of the transmitters is considered. The movement model for the transmitters corresponding to Eq. (2.12) is expressed as

$$\boldsymbol{x}_{\mathrm{TX},k} = \boldsymbol{x}_{\mathrm{TX},k-1} + \boldsymbol{w}_{\mathrm{TX},k-1}, \tag{3.19}$$

where $\boldsymbol{w}_{\mathrm{TX},k-1}$ is a noise sample from a known zero-mean distribution with very small variance from which sampling is possible.

For modeling the movement of a pedestrian, various models exist in the literature [BLK04]. In Channel-SLAM, a gyroscope, for example from a smartphone, is incorporated into a Gaussian transition model [BLK04] as control input [Gen18]. The heading change rate $\Delta_{\beta,k}$ is the difference in the heading of the user between time instants k and $k-1$ as measured by the gyroscope. At time instant k, the relative rotation matrix $\boldsymbol{R}(\Delta_{\beta,k})$ is defined as

$$\boldsymbol{R}(\Delta_{\beta,k}) = \begin{bmatrix} \cos\left(\Delta_{\beta,k} + \omega_k\right) & -\sin\left(\Delta_{\beta,k} + \omega_k\right) \\ \sin\left(\Delta_{\beta,k} + \omega_k\right) & \cos\left(\Delta_{\beta,k} + \omega_k\right) \end{bmatrix}, \tag{3.20}$$

where ω_k is a heading noise sample following a von Mises distribution [JS01] to model inaccuracies in the gyroscope measurements. With the rotation matrix, the movement model of the user corresponding to Eq. (2.12) is given by

$$\boldsymbol{x}_{\mathrm{u},k} = \underbrace{\begin{bmatrix} \mathbb{1}_2 & T_k \mathbb{1}_2 \\ \mathbb{0}_2 & \boldsymbol{R}(\Delta_{\beta,k}) \end{bmatrix}}_{\boldsymbol{F}_{\mathrm{u},k}} \boldsymbol{x}_{\mathrm{u},k-1} + \boldsymbol{w}_{\mathrm{u},k-1} = \boldsymbol{F}_{\mathrm{u},k} \boldsymbol{x}_{\mathrm{u},k-1} + \boldsymbol{w}_{\mathrm{u},k-1}, \tag{3.21}$$

where T_k is the time between time instants $k-1$ and k, and $\boldsymbol{w}_{\mathrm{u},k-1}$ a noise sample from a known zero-mean distribution as in [RAG04]. The speed of the user is assumed constant in Eq. (3.21). Changes in the speed are covered by the covariance matrix of the noise sample $\boldsymbol{w}_{\mathrm{u},k-1}$ [Gen18]. An overview of the involved state variables and measurements is provided in Fig. 3.10.

The measurement noise is assumed to be zero-mean Gaussian with variances ${\sigma_{\tau,k}^{<j>}}^2$ and ${\sigma_{\theta,k}^{<j>}}^2$ for the ToA and AoA measurements, respectively. These variances are obtained from the covariance estimates of KEST. In the update step of the user particle filter, the likelihood of the i^{th} user particle regarding the ℓ^{th} particle of the j^{th} transmitter is calculated as

$$\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k} | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right) = \mathcal{N}\left(\tau_{j,k}; \hat{\tau}_k^{<i,j,\ell>}, {\sigma_{\tau,k}^{<j>}}^2\right) \mathcal{N}\left(\theta_{j,k}; \hat{\theta}_k^{<i,j,\ell>}, {\sigma_{\theta,k}^{<j>}}^2\right) \tag{3.22}$$

with the predicted measurements

$$\hat{\tau}_k^{<i,j,\ell>} = \frac{1}{c_0} \| \boldsymbol{p}_{\mathrm{u},k}^{<i>} - \boldsymbol{p}_{\mathrm{TX},k}^{<i,j,\ell>} \| + \tau_{0,k}^{<i,j,\ell>} \tag{3.23}$$

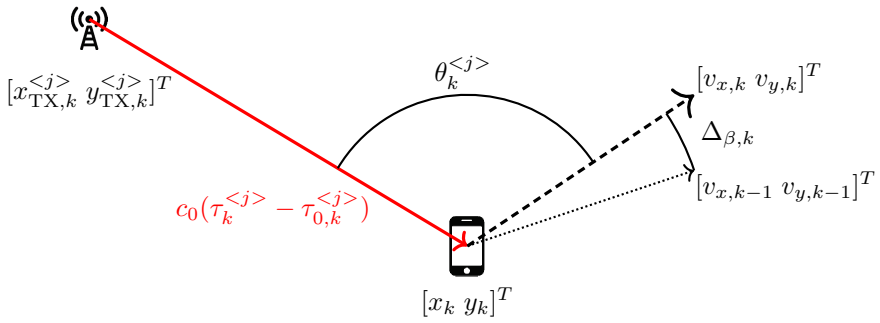


Figure 3.10: At time instant k, the user location and velocity are $[x_k\ \ y_k]^T$ and $[v_{x,k}\ \ v_{y,k}]^T$, respectively. The heading change of $\Delta_{\beta,k}$ is relative to the velocity at time instant $k-1$, $[v_{x,k-1}\ \ v_{y,k-1}]^T$. The signal from the j^{th} transmitter located at $[x_{\mathrm{TX},k}^{<j>}\ \ y_{\mathrm{TX},k}^{<j>}]^T$ is drawn red. It arrives at the user with a ToA $\tau_k^{<j>} - \tau_{0,k}^{<j>}$ at an angle of $\theta_k^{<j>}$ relative to the user heading.

for the ToA following Eq. (2.2), and

$$\hat{\theta}_k^{<i,j,\ell>} = \operatorname{atan2}\left(y_{\mathrm{TX},k}^{<i,j,\ell>} - y_k^{<i>}, x_{\mathrm{TX},k}^{<i,j,\ell>} - x_k^{<i>}\right) - \operatorname{atan2}\left(v_{y,k}^{<i>}, v_{x,k}^{<i>}\right) \tag{3.24}$$

for the AoA following Eq. (2.4), respectively. In Eqs. (3.23) and (3.24),

- $\boldsymbol{p}_{\mathrm{u},k}^{<i>} = \left[x_k^{<i>}\ \ y_k^{<i>}\right]^T$ is the location and
- $\left[v_{y,k}^{<i>}\ \ v_{x,k}^{<i>}\right]^T$ the velocity of the i^{th} user particle, and
- $\boldsymbol{p}_{\mathrm{TX},k}^{<i,j,\ell>} = \left[x_{\mathrm{TX},k}^{<i,j,\ell>}\ \ y_{\mathrm{TX},k}^{<i,j,\ell>}\right]^T$ is the location and
- $\tau_{0,k}^{<i,j,\ell>}$ the clock offset of the ℓ^{th} transmitter particle.

The importance density in the user particle filter is chosen to be the state transition prior,

$$\mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},k} | \boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) := \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k} | \boldsymbol{x}_{\mathrm{u},k-1}, \boldsymbol{u}_k\right), \tag{3.25}$$

which is obtained from the movement model in Eq. (3.21). The weights of the user particle filters are then calculated recursively as [Gen+16a]

$$w_k^{<i>} \propto w_{k-1}^{<i>} \prod_{j=1}^{N_{\mathrm{TX},k}} \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{k-1}^{<i,j,\ell>}\, \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k} | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right), \tag{3.26}$$

where $\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k} | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right)$ is calculated as in Eq. (3.22). It follows directly from Eq. (3.26) that the weight updates in the transmitter particle filters are

$$w_k^{<i,j,\ell>} \propto w_{k-1}^{<i,j,\ell>}\, \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k} | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right). \tag{3.27}$$

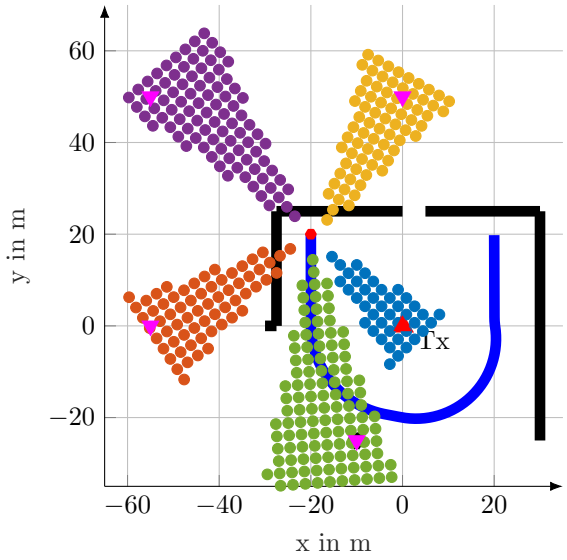


Figure 3.11: The plot shows the estimates of the transmitter locations after initializing five transmitters in the first time instant of Channel-SLAM. Each location estimate is represented as a particle cloud. A particle cloud with a different color corresponds to a different transmitter. The red triangle is the location of the physical transmitter, and the magenta downward triangles are ground truth locations of virtual transmitters. The red circle is the user position.

After the update of the user particle filter, the weights of all transmitter particle filters and the user particle filters are normalized to one. Resampling is performed in all particle filters if necessary.

When KEST detects a new signal component, a new transmitter is initialized. A new transmitter particle filter is created for each of the user particles. The initial state PDF is based on the KEST estimates for the mean and the covariance matrix of the signal component's parameters, i.e., the measurements for the corresponding transmitter. Fig. 3.11 visualizes the initialization of five transmitters in Channel-SLAM at the first time instant for the scenario in Fig. 3.3. The red triangle labeled Tx marks the location of the physical transmitter, and the magenta triangles pointing down are the true locations of the detected virtual transmitters. The location of the user is depicted by the red circle at the beginning of the user track drawn in blue. There are five particle clouds in different colors, corresponding to location estimates for the five transmitters. Both ToA and AoA measurements are assumed to be available.

As mentioned above and becomes evident in Fig. 3.11, even if both ToA and AoA measurements are available, the system considered at one time instant is underdetermined, leading to a high uncertainty about a new transmitter's state. Over time, as measurements for this transmitters are taken from different user positions, the transmitter's state uncertainty tends to shrink, and the trans-

mitter provides more and more information about the user state. Thus, signal components that are tracked over a long distance, are preferable. In contrast, transmitters with a short life span do not contribute much to the positioning performance of Channel-SLAM, but only increase the computational complexity. When KEST loses track of a signal component, the corresponding transmitter particle filters are removed for all user particles.

The Channel-SLAM algorithm for one time instant $k > 0$ without initializing or discarding transmitters is presented in Algorithm 3.

Algorithm 3: Channel-SLAM algorithm at one time instant $k > 0$.

Input:
user particles $\boldsymbol{x}_{\mathrm{u},0:k-1}^{<i>}$ and weights $w_{0:k-1}^{<i>}$ for $i = 1, ..., N_p$,
transmitter particles $\boldsymbol{x}_{\mathrm{u},0:k-1}^{<i,j,\ell>}$ and weights $w_{0:k-1}^{<i,j,\ell>}$ for $i = 1, ..., N_p$, $j = 1, ..., N_{\mathrm{TX},k}^{\mathrm{a}}$,$\ell = 1, ..., N_{p,\mathrm{Tx}}$,
measurement $\boldsymbol{z}_{\mathrm{R},k}$ and control input $\boldsymbol{u}_k$

Output:
user particles $\boldsymbol{x}_{\mathrm{u},k}^{<i>}$ and weights $w_k^{<i>}$ for $i = 1, ..., N_p$,
transmitter particles $\boldsymbol{x}_{\mathrm{u},k}^{<i,j,\ell>}$ and weights $w_k^{<i,j,\ell>}$ for $i = 1, ..., N_p$, $j = 1, ..., N_{\mathrm{TX},k}^{\mathrm{a}}$,$\ell = 1, ..., N_{p,\mathrm{Tx}}$,

for $i = 1, ..., N_p$ **do**
  draw a new user particle $\boldsymbol{x}_{\mathrm{u},k}^{<i>}$ from $\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k} | \boldsymbol{x}_{\mathrm{u},k-1}^{<i>}, \boldsymbol{u}_k\right)$;
  for $j = 1, ..., N_{TX,k}$ **do**
    for $\ell = 1, ..., N_{p,Tx}$ **do**
      draw a new transmitter particle $\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}$ from $\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k} | \boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j,\ell>}\right)$;
      calculate the weight $w_k^{<i,j,\ell>}$ with Eq. (3.27);
  calculate the weight $w_k^{<i>}$ with Eq. (3.26);
normalize all user and transmitter weights with Eq. (2.29) ;
if *any user or transmitter particles suffer from degeneracy* **then**
  resample the corresponding particles

A dynamic Bayesian network of Channel-SLAM is depicted in Fig. 3.12. A first order Markov process is assumed. The quantities at time instant k are surrounded by the dashed rectangle. The transmitters' state $\boldsymbol{x}_{\mathrm{TX},k}$ and the user state $\boldsymbol{x}_{\mathrm{u},k}$ determine the measurements $\boldsymbol{z}_{\mathrm{R},k}$. If the user state at time instant $k-1$ is known, the user states prior to time instant $k-1$ are not relevant for the knowledge of the user state at time instant k. The same holds for the transmitters' state.

3.6 Extensions to Channel-SLAM

Channel-SLAM does not require certain measurement types such as ToA or AoA. In principle, any kind of measurement relating the state of the user to the

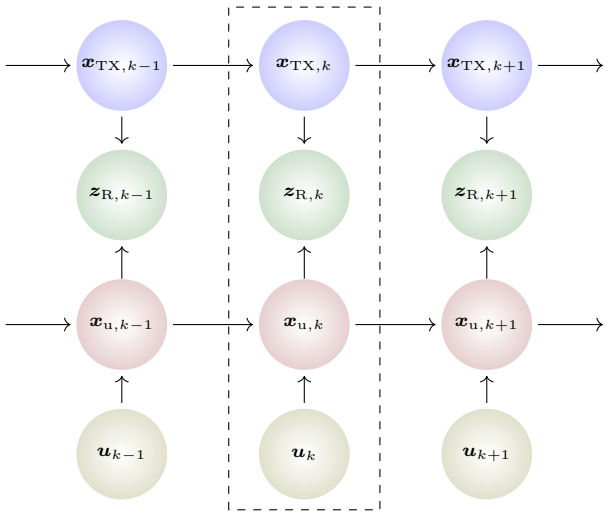


Figure 3.12: The dynamic Bayesian network illustrates the conditional dependencies among the involved variables in Channel-SLAM. A Markov process of order one is assumed.

states of the transmitters can be used. For example, in [GJD13b] or [Ulm+15], only time differences of arrival (TDoAs) among signal components are considered. Thus, no time synchronization between the physical transmitter and the user is required. In [Gen+14], the Doppler shift is exploited to gain information on the AoA of signal components. The likelihood in Eq. (3.22) needs to be adapted to the kinds of measurement used. More different kinds of measurements lead to a better positioning performance of Channel-SLAM, and a shorter convergence time of the particle filters.

In addition to measurements relating the user to the transmitter states, additional sensors can be used in the particle filter of Channel-SLAM. In the particle filter above, gyroscope measurements are incorporated for the prediction of the user movement. In [Gen+17a], the authors have also included GNSS raw measurements into the particle filter for positioning on a global scale.

To improve the predictability of the user movement, location-based maps as in [AR12] can be used. Such hexagonal grid maps that store the history of a user trajectory are incorporated into Channel-SLAM in [Gen+16c]. Furthermore, in [Gen+16b], hexagonal grid maps are used to detect and exploit loop closures in Channel-SLAM. Loop closures occur when the user revisits a location.

In a practical implementation of a particle filter, the weights of some particles have values close to zero after the update step. To improve the numerical stability of Channel-SLAM in such a case, the weights can be represented in a logarithmic domain. A Channel-SLAM algorithm that works completely in a logarithmic domain is presented in [GZJ18].

As a user is in motion and takes measurements of transmitters from different locations, the initially high uncertainties about the transmitters' states tend to shrink. The smaller the uncertainty, the fewer particles are necessary for a good approximation. In [Gen+17b], the authors have presented a particle reduction method for the transmitter state estimates. The transmitter state space is divided into a discrete grid of bins. In each bin, there is only a certain number of particles allowed. When the particles are resampled and a bin is already fully occupied with particles, any further particle that would be resampled into that bin is deleted. The weight of the deleted particles is proportionately added to the weights of the particles that are already in the corresponding bin.

3.7 Summary and Outlook

The Channel-SLAM algorithm is the foundation for the work performed in this thesis. It is a multipath assisted positioning algorithm that works in two steps. In the first step, a channel estimator tracks the parameters of signal components. In the second step, these estimates are used to jointly localize a single user and map physical and virtual transmitters.

SLAM has mainly been studied in robotics, where a mobile robot position is estimated simultaneously with mapping the environment. A map of the environment consists in general of a set of landmarks. A fundamental difference of SLAM in robotics and in multipath assisted positioning is the nature of these landmarks. In robotic SLAM schemes, visual sensors such as cameras, or radar based sensors are used widely. The estimated maps are therefore maps of actual physical objects detected by the robot, such as walls, doors, or furniture, for example, resulting in a physical plan of the environment. When features of these objects, for example structures of objects or visual patterns, can be extracted, they can be used to differentiate among them [Low+16]. In multipath assisted positioning, though, states of physical and virtual radio transmitters are mapped. Virtual transmitters have no correspondence in the physical world. They merely describe locations where MPCs seem to originate from. Since virtual transmitters arise only due to reflections and scattering of a transmit signal, it is hard to differentiate among them based on the features of the signal, such as waveform or transmit signal for example.

The next chapter builds upon Channel-SLAM, and augments the transmitter states by visibility information. As we will see, visibility information by itself increases the positioning performance of Channel-SLAM already. In addition, it can be exploited for data association and is a key enabler for cooperation among users in Channel-SLAM, since it facilitates differentiating among transmitters.

Chapter 4

Channel-SLAM with Visibility Mapping

In Channel-SLAM, the states of the transmitters are estimated jointly with the user state. The estimates of the transmitter states are denoted by the term transmitter state map, or just state map. To improve the positioning performance, state maps can be augmented by additional information about transmitters. In the following, state maps are augmented by the information from which user positions a transmitter is *visible*. A transmitter is visible from a position if the user can receive the transmit signal in a LoS condition. The estimates of visibilities of transmitters are denoted by the term visibility map.

In this chapter, we derive a new tracking filter for Channel-SLAM that estimates not only the states but also visibility maps of transmitters. Estimating and exploiting such visibility maps increases the positioning performance of single user Channel-SLAM, as it increases the robustness of detecting loop closures. When a loop closure is detected, uncertainties in the user and the transmitter state estimates shrink. If a user revisits a location where it had been before, the same set of transmitters is likely to be visible again.

Data association is of crucial importance in any SLAM scheme. In Channel-SLAM, it refers to associating detected signal components with transmitters across time instants. In multipath assisted positioning, it is difficult to differentiate among transmitters, since their state comprises only their location and clock offset. In that sense, the diversity among transmitters is low, which makes data association difficult in Channel-SLAM. Augmenting the transmitter states by visibility information increases this diversity as it provides an additional source of information to differentiate among transmitters. Being able to reliably differentiate among transmitters enables a robust data association.

The definition of a visibility map is presented in Section 4.1. In Section 4.2, a new Rao-Blackwellized particle filter for Channel-SLAM including the estimation and exploitation of visibility maps is derived. A data association scheme for the new particle filter is derived in Section 4.3.

4.1 Visibility Maps

A transmitter might be visible from the same user position at one time instant, but not at another. Some of the possible reasons for this effect are the following.

- Low received power of the signal components: even when a transmitter is in LoS, the channel estimator might be able or unable to track the corresponding signal component depending on the received power of the signal component and the receiver sensitivity. For example, the direction from which the user is coming or small scale fading may play a role.
- Changes in the environment: objects or people temporarily block certain propagation paths or cause additional multipath propagation.
- Data association: if a signal component gets associated with the wrong transmitter, that transmitter may be falsely deemed visible or not visible.
- Biased user state estimate: if the user position is estimated with a bias, the visibility estimates may be biased as well.

To cope with these issues, information on transmitter visibilities is stored in a location-based, probabilistic map for a convenient representation of the spatial visibility information and for a simple mapping process. To create such a map, the two-dimensional space is divided into hexagons. A two-dimensional space can be tiled with only three shapes of equal side lengths, namely triangles, squares and hexagons. Hexagons are chosen to minimize the discretization error, since the distance of the center of the corresponding shape to an arbitrary point within the shape is on average shorter for hexagons compared to squares and triangles of equal area [BOB07]. In the h^{th} hexagon, the probability that the j^{th} transmitter is visible from an arbitrary position within the hexagon is denoted by $V_h^{<j>}$. The probability that the transmitter is not visible is $\bar{V}_h^{<j>} = 1 - V_h^{<j>}$.

An example for such a visibility map regarding one physical transmitter Tx is depicted in Fig. 4.1. The black lines are walls that reflect or block the transmit signal. The color of a hexagon denotes the probability that the transmitter Tx is in LoS from an arbitrary position within this hexagon. For example, in the green hexagons, this probability equals one, whereas in the red hexagons, this probability equals zero as the red hexagons are shadowed by walls. In the yellow hexagons, the probability is 1/2, corresponding to the probability to be in front of or behind the walls dividing the hexagons into two parts of equal size.

Visibility information regarding one hexagon comprises the probabilities of each of the N_{TX}^{a} transmitters being visible from within the hexagon. The visibility information for the h^{th} hexagon is therefore the set of probabilities

$$M_h = \left\{ V_h^{<1>}, \ldots, V_h^{<N_{\text{TX}}^{\text{a}}>} \right\}. \tag{4.1}$$

The visibility probabilities for all N_H hexagons are comprised in the set

$$M = \{ M_1, \ldots, M_{N_H} \}. \tag{4.2}$$

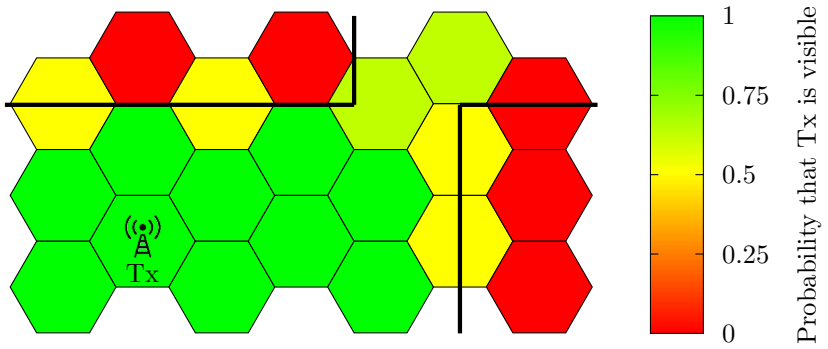


Figure 4.1: The LoS signals from the physical transmitter Tx can be received only at certain locations in the scenario due to blocking by walls. The colors of the hexagons indicate the probability that the transmitter is in LoS from an arbitrary location within the hexagon.

In a nutshell, a visibility map M comprises the probabilities of each of the $N_{\mathrm{TX}}^{\mathrm{a}}$ transmitters being visible from each of the N_H hexagons. However, the true probabilities $V_h^{<j>}$ of the single transmitters being visible in the hexagons are unknown. The corresponding random variable at time instant k is denoted by $\boldsymbol{M}_{h,k}^{<j>}$. Analogue to Eqs. (4.1) and (4.2), the set $\boldsymbol{M}_{h,k}$ comprises the random variables $\boldsymbol{M}_{h,k}^{<j>}$ for all transmitters in the hexagon with index h, and the set $\boldsymbol{M}_k$ comprises the random variables $\boldsymbol{M}_{h,k}^{<j>}$ for all transmitters and all hexagons.

The measurements by a user regarding the visibilities of the transmitters from time instants one to k are denoted by $\boldsymbol{z}_{\mathrm{V},1:k}$. Section 4.2.1 deals with the question how to obtain $\boldsymbol{z}_{\mathrm{V},1:k}$. With these measurements, the conditional PDFs of the random variables $\boldsymbol{M}_{h,k}^{<j>}$, namely $\mathrm{p}\left(\boldsymbol{M}_{h,k}^{<j>} | \boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{z}_{\mathrm{V},1:k}\right)$, can be estimated. Each of these PDFs is assumed to follow a Beta distribution, which represents the degree of belief at time instant k that the j^{th} transmitter is visible in the h^{th} hexagon. The PDF of a Beta distribution with positive parameters p and q is defined as

$$\mathcal{B}\left(x; p, q\right) = \begin{cases} \frac{1}{\mathrm{B}(p,q)} x^{p-1}(1-x)^{q-1} & \text{if } 0 < x < 1 \\ 0 & \text{otherwise} \end{cases}, \tag{4.3}$$

where the Beta function

$$\mathrm{B}\left(p, q\right) = \frac{\Gamma\left(p\right)\Gamma\left(q\right)}{\Gamma\left(p+q\right)} \tag{4.4}$$

is defined by means of the Gamma function $\Gamma\left(\cdot\right)$ [AAR99]. The PDF of the Beta distribution is plotted in Fig. 4.2 exemplarily for different parameters. Generally, the greater the sum of the parameters, the sharper is the peak of the PDF.

The parameters of the Beta distributions representing the belief that the j^{th} transmitter is visible at time instant k from the h^{th} hexagon are denoted by

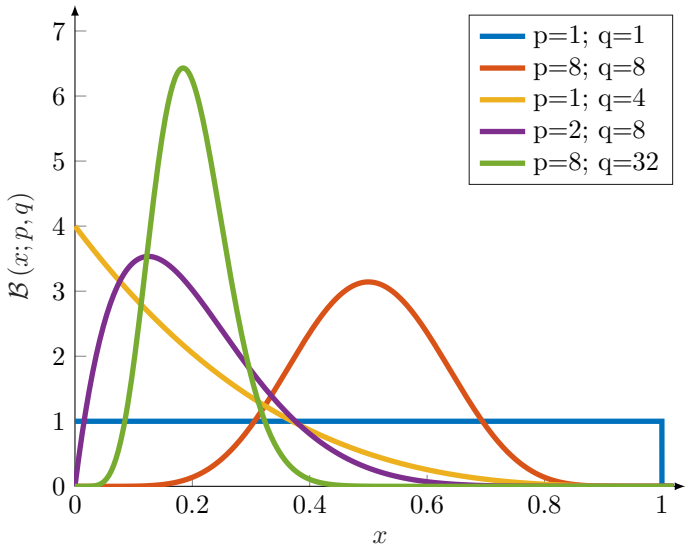


Figure 4.2: The PDF of the Beta distribution is plotted for different parameters.

$p_{h,k}^{<j>}$ and $q_{h,k}^{<j>}$. Both of the parameters $p_{h,k}^{<j>}$ and $q_{h,k}^{<j>}$ are the sum of two addends, namely

$$\begin{aligned} p_{h,k}^{<j>} &= C_{h,k}^{<j>} + \nu_h^{<j>}, \quad \text{and} \\ q_{h,k}^{<j>} &= \bar{C}_{h,k}^{<j>} + \bar{\nu}_h^{<j>}. \end{aligned} \tag{4.5}$$

The first addends, $C_{h,k}^{<j>}$ and $\bar{C}_{h,k}^{<j>}$, are based on the user observations $\boldsymbol{z}_{\mathrm{V},1:k}$ and are thus dependent on the time instant k. To learn a visibility map, these addends are constantly updated by the user. In Section 4.2.3 we will explain how to obtain them.

The second addends, $\nu_h^{<j>}$ and $\bar{\nu}_h^{<j>}$, correspond to prior knowledge and are therefore independent from the time instant. They have two purposes. On the one hand, they can prevent overly confident conclusions about the corresponding transmitter's visibility if only few or no observations are available. In such a case, $C_{h,k}^{<j>}$ and $\bar{C}_{h,k}^{<j>}$ have very small values or equal zero. On the other hand, they can be used to incorporate prior knowledge about the visibilities. If no such prior knowledge is available, $\nu_h^{<j>} = \bar{\nu}_h^{<j>}$. The possibility to include such prior knowledge to the user observations makes the Beta distribution suitable for the phase of learning a visibility map. The number of user observations can then be regarded as a measure of trust in the belief in the visibilities of the transmitters.

Finally, the PDF $\mathrm{p}\left(\boldsymbol{M}_{h,k}^{<j>} | \boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{z}_{\mathrm{V},1:k}\right)$ representing the belief that the j^{th} transmitter is visible in the h^{th} hexagon is expressed with $\bar{\boldsymbol{M}}_{h,k}^{<j>} = 1 - \boldsymbol{M}_{h,k}^{<j>}$

as the Beta distribution

$$\begin{aligned} \mathrm{p}\left(\boldsymbol{M}_{h,k}^{<j>} | \boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{z}_{\mathrm{V},1:k}\right) &= \mathcal{B}\left(\boldsymbol{M}_{h,k}^{<j>}; p_{h,k}^{<j>}, q_{h,k}^{<j>}\right) \\ &= \frac{1}{\mathrm{B}\left(p_{h,k}^{<j>}, q_{h,k}^{<j>}\right)} \left(\boldsymbol{M}_{h,k}^{<j>}\right)^{p_{h,k}^{<j>}-1} \left(\bar{\boldsymbol{M}}_{h,k}^{<j>}\right)^{q_{h,k}^{<j>}-1}. \end{aligned} \tag{4.6}$$

At time instant k, the expectation values of the Beta distribution for the j^{th} transmitter in the h^{th} hexagon to be visible and not visible are denoted by

$$\mathbb{E}_{h,k}^{<j>} := \mathbb{E}\left[\mathcal{B}\left(\boldsymbol{M}_{h,k}^{<j>}; p_{h,k}^{<j>}, q_{h,k}^{<j>}\right)\right] = \frac{p_{h,k}^{<j>}}{p_{h,k}^{<j>} + q_{h,k}^{<j>}} \tag{4.7}$$

and

$$\bar{\mathbb{E}}_{h,k}^{<j>} := 1 - \mathbb{E}_{h,k}^{<j>} = \frac{q_{h,k}^{<j>}}{p_{h,k}^{<j>} + q_{h,k}^{<j>}}, \tag{4.8}$$

respectively.

The memory requirement of a visibility map estimate in amount of numbers to be stored is $2N_H N_{\mathrm{TX}}^{\mathrm{a}}$, corresponding to the two parameters $p_{h,k}^{<j>}$ and $q_{h,k}^{<j>}$ for each observed transmitter and each hexagon in the map.

4.2 Particle Filter for Channel-SLAM using Visibility Information

Visibility information can be used in a Rao-Blackwellized particle filter similar to the one in Channel-SLAM. The user location is estimated simultaneously with mapping the states and visibilities of transmitters. If a user comes across a location a second time, which is denoted by the term loop closure, the visibility estimates help correcting the user location estimate. In the original Channel-SLAM algorithm in Chapter 3, no use is made of loop closures.

4.2.1 Visibility Measurements

In Section 3.5, the vector $\boldsymbol{z}_{\mathrm{R},k}$ of radio measurements was obtained from the inner stage of KEST and contained the ToAs and AoAs of all the $N_{\mathrm{TX},k}$ currently visible transmitters as in Eq. (3.8). It was assumed that the associations of signal components from one time instant to another were known implicitly. In the following, the additional measurement vector $\boldsymbol{z}_{\mathrm{V},k}$ is introduced making these associations explicit. The vector $\boldsymbol{z}_{\mathrm{V},k}$ is of length $N_{\mathrm{TX},k}^{\mathrm{a}}$ and its j^{th} entry is denoted by $z_{\mathrm{V},k}^{<j>}$. If the j^{th} of the $N_{\mathrm{TX},k}^{\mathrm{a}}$ transmitters is visible at time instant k, then $z_{\mathrm{V},k}^{<j>}$ is the index of the ToA measurement for that transmitter in $\boldsymbol{z}_{\mathrm{R},k}$. Otherwise, $z_{\mathrm{V},k}^{<j>}$ is zero. Thus, $\boldsymbol{z}_{\mathrm{V},k}$ has $N_{\mathrm{TX},k}$ non-zero entries.

Fig. 4.3 illustrates the relation between $\boldsymbol{z}_{\mathrm{R},k}$ and $\boldsymbol{z}_{\mathrm{V},k}$ exemplarily for the case that only ToA radio measurements are used. The $N_{\mathrm{TX},k} = 3$ of the $N_{\mathrm{TX},k}^{\mathrm{a}} = 8$

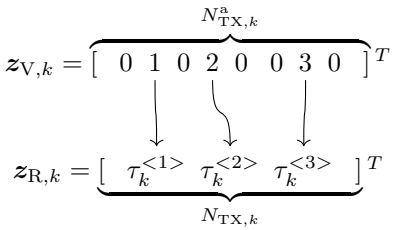


Figure 4.3: The j^{th} entry of $\boldsymbol{z}_{\mathrm{V},k}$ denotes the index of the ToA measurement for the j^{th} of the $N^{\mathrm{a}}_{\mathrm{TX},k}$ transmitters in $\boldsymbol{z}_{\mathrm{R},k}$ at time instant k, if that j^{th} transmitter is visible. If the j^{th} transmitter is not visible, this j^{th} entry is zero.

transmitters that are visible at time instant k have the indices two, four and seven, corresponding to the indices of the non-zero entries in $\boldsymbol{z}_{\mathrm{V},k}$. For example, the first entry in $\boldsymbol{z}_{\mathrm{R},k}$ is the ToA measurement of the second of the $N^{\mathrm{a}}_{\mathrm{TX},k} = 8$ transmitters that have been visible so far. I.e., it is the ToA estimate of the second of the $N^{\mathrm{a}}_{\mathrm{TX},k} = 8$ signal components that have been tracked by KEST from time instants one to k. If AoA measurements are available as well, the length of $\boldsymbol{z}_{\mathrm{R},k}$ is $2N_{\mathrm{TX},k}$ as in Eq. (3.8), but the vector $\boldsymbol{z}_{\mathrm{V},k}$ stays the same.

Hence, the history $\boldsymbol{z}_{\mathrm{V},1:k}$ holds information on associations among signal components, or transmitters, at consecutive time instants. At time instant k, $\boldsymbol{z}_{\mathrm{V},k}$ indicates which of the $N^{\mathrm{a}}_{\mathrm{TX},k}$ transmitters are visible. Thus, it is denoted by the term visibility measurement vector. The visibility measurement vector is obtained from KEST. No additional actual measurements are necessary.

The measurement vector $\boldsymbol{z}_k$ comprises the radio measurement vector $\boldsymbol{z}_{\mathrm{R},k}$ and the visibility measurement vector $\boldsymbol{z}_{\mathrm{V},k}$,

$$\boldsymbol{z}_k = \left[{\boldsymbol{z}_{\mathrm{R},k}}^T \ \ {\boldsymbol{z}_{\mathrm{V},k}}^T \right]^T . \tag{4.9}$$

4.2.2 Structure of the Particle Filter

By including the visibility measurement vector into the measurements as in Eq. (4.9), both the states $\boldsymbol{x}_{\mathrm{TX},0:k}$ and the visibility maps $\boldsymbol{M}_{0:k}$ can be estimated jointly with the user states in Channel-SLAM. Fig. 4.4 shows the corresponding dynamic Bayesian network. Comparing Fig. 4.4 to Fig. 3.12, the dynamic Bayesian network in Fig. 4.4 is expanded by the visibility map $\boldsymbol{M}$ and the visibility measurement vector $\boldsymbol{z}_{\mathrm{V}}$.

The new posterior PDF that is to be estimated is

$$\begin{aligned}
&\mathrm{p}\left(\boldsymbol{x}_{0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \\
&\quad = \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}, \boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \\
&\quad = \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k} | \boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \\
&\quad = \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k} | \boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{z}_{1:k}\right),
\end{aligned} \tag{4.10}$$

exploiting that the transmitter states $\boldsymbol{x}_{\mathrm{TX},0:k}$ are conditionally independent from

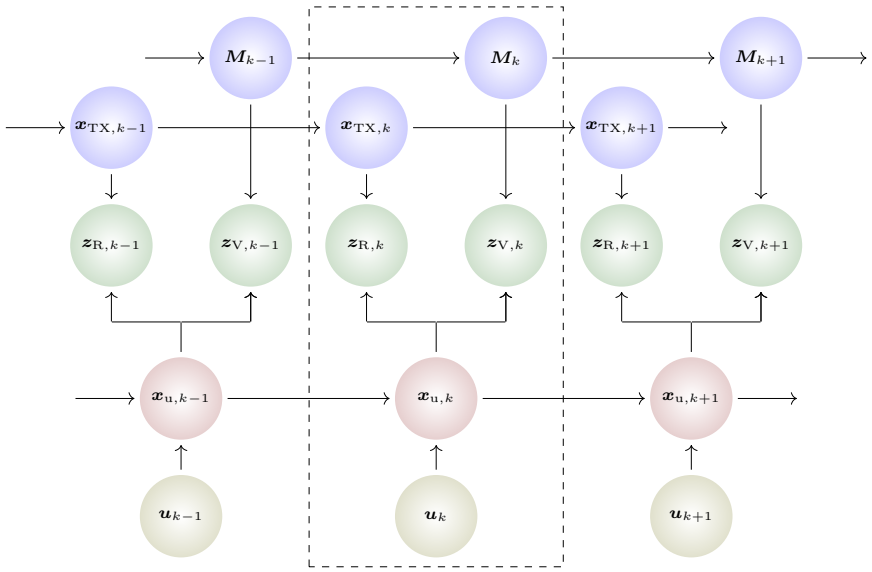


Figure 4.4: The dynamic Bayesian network illustrates the conditional dependencies among the involved variables in Channel-SLAM with visibility information on transmitters.

the control input $\boldsymbol{u}_{1:k}$ and assuming that the transmitter states do not depend on the visibility maps $\boldsymbol{M}_{0:k}$.

Following the structure in Eq. (4.10), a Rao-Blackwellized particle filter is used to jointly estimate the user states and the visibility maps on the one hand, and the transmitter states on the other hand. The particle filter estimating the user states and the transmitter visibility maps is denoted by the term user-map particle filter, and a particle in that filter a user-map particle. Fig. 4.5 illustrates the structure of the Rao-Blackwellized particle filter. Each user-map particle carries a hypothesis of the transmitter states and the transmitter visibility map. The i^{th} user-map particle history at time instant k is of the form

$$[\boldsymbol{x}_{\mathrm{u},0:k} \;\; \boldsymbol{M}_{0:k}]^{<i>} := \left[\boldsymbol{x}_{\mathrm{u},0:k}^{<i>} \;\; \boldsymbol{M}_{0:k}^{<i>}\right], \tag{4.11}$$

comprising the history of the user particle $\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}$ and the visibility map $\boldsymbol{M}_{0:k}^{<i>}$. Regarding the Rao-Blackwellized particle filter, the information for the i^{th} user-map particle comprises

$$\left\langle [\boldsymbol{x}_{\mathrm{u},0:k} \;\; \boldsymbol{M}_{0:k}]^{<i>}, \left\{ \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{R},1:k}\right)\right\}_{j=1,\ldots,N_{\mathrm{TX},k}^{\mathrm{a}}} \right\rangle. \tag{4.12}$$

Analogue to Eq. (3.16), the first factor in the last line of Eq. (4.10), the posterior PDF of the user states and the visibility maps, is expressed as a

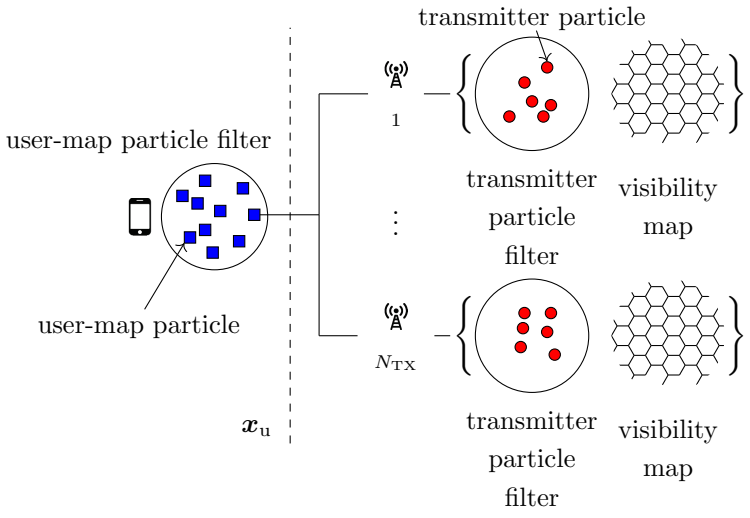


Figure 4.5: The structure of the Rao-Blackwellized particle filter for Channel-SLAM with visibility information. The posterior PDFs of the transmitter states are represented by particle clouds for every particle in the user-map particle filter. A hexagonal visibility map is estimated jointly with every user-map particle.

weighted sum of particles,

$$\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) = \sum_{i=1}^{N_p} w_{0:k}^{<i>} \delta\left(\left[\boldsymbol{x}_{\mathrm{u},0:k} \;\; \boldsymbol{M}_{0:k}\right] - \left[\boldsymbol{x}_{\mathrm{u},0:k} \;\; \boldsymbol{M}_{0:k}\right]^{<i>}\right), \tag{4.13}$$

where $w_{0:k}^{<i>}$ is the weight of the i^{th} user-map particle.

In the Rao-Blackwellized particle filter, each transmitter's state is estimated for every user-map particle independently from the other transmitters' states as in the particle filter in Section 3.5. The posterior PDF of the transmitter states given the i^{th} user-map particle is

$$\begin{aligned} \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i>} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{1:k}\right) &= \prod_{j=1}^{N_{\mathrm{TX},k}} \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{1:k}\right) \\ &= \prod_{j=1}^{N_{\mathrm{TX},k}} \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{0:k}^{<i,j,\ell>} \delta\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>} - \boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j,\ell>}\right), \end{aligned} \tag{4.14}$$

where $\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j,\ell>}$ is the history of the ℓ^{th} particle in the transmitter particle filter for the j^{th} transmitter of the i^{th} user-map particle and $w_{0:k}^{<i,j,\ell>}$ its associated weight.

4.2.3 Weight Derivation

In the user-map particle filter, samples need to be drawn from the posterior PDF on the left hand side of Eq. (4.13). In general, it is hard or computationally expensive to draw such samples. Instead, the idea of importance sampling can be applied as in Section 2.2.3. With importance sampling, samples are drawn from an importance density $\mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right)$. The mismatch between sampling from the importance density rather than from the posterior PDF is compensated by the weights on the right hand side of Eq. (4.13). If the posterior PDF can be evaluated, the weights are calculated by [Aru+02]

$$w_k^{<i>} = \frac{\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{M}_{0:k}^{<i>} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right)}{\mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{M}_{0:k}^{<i>} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right)}. \tag{4.15}$$

To calculate the weights in Eq. (4.15), the entire history of the user state and the visibility map has to be considered. However, with the assumption of a first order Markov process and a causal system, the importance density can be factorized as

$$\begin{aligned} \mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) = \mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},k}, \boldsymbol{M}_k | \boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{M}_{0:k-1}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \\ \times \mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{M}_{0:k-1} | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k-1}\right), \end{aligned} \tag{4.16}$$

which allows to calculate the weights recursively up to proportionality by

$$\begin{aligned} w_k^{<i>} &\propto \mathrm{p}\left(\boldsymbol{z}_k | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{M}_{0:k}^{<i>}, \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right) \\ &\quad \times \frac{\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_k^{<i>} | \boldsymbol{x}_{\mathrm{u},k-1}^{<i>}, \boldsymbol{M}_{k-1}^{<i>}, \boldsymbol{u}_k\right)}{\mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_k^{<i>} | \boldsymbol{x}_{\mathrm{u},0:k-1}^{<i>}, \boldsymbol{M}_{0:k-1}^{<i>}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right)} \\ &\quad \times \frac{\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k-1}^{<i>}, \boldsymbol{M}_{0:k-1}^{<i>} | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k-1}\right)}{\mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},0:k-1}^{<i>}, \boldsymbol{M}_{0:k-1}^{<i>} | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k-1}\right)} \end{aligned} \tag{4.17}$$

$$\begin{aligned} &= w_{k-1}^{<i>} \, \mathrm{p}\left(\boldsymbol{z}_k | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{M}_{0:k}^{<i>}, \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right) \\ &\quad \times \frac{\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_k^{<i>} | \boldsymbol{x}_{\mathrm{u},k-1}^{<i>}, \boldsymbol{M}_{k-1}^{<i>}, \boldsymbol{u}_k\right)}{\mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_k^{<i>} | \boldsymbol{x}_{\mathrm{u},0:k-1}^{<i>}, \boldsymbol{M}_{0:k-1}^{<i>}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right)}. \end{aligned} \tag{4.18}$$

The details of the derivation of Eq. (4.18) for the user-map particle filter is presented in Appendix B.1. The derivation in a general form can be found in [Aru+02].

The radio measurement $\boldsymbol{z}_{\mathrm{R}}$ is independent from any visibility information or control input. In addition, the visibility measurement $\boldsymbol{z}_{\mathrm{V}}$ depends only on the visibility map $\boldsymbol{M}$ and the user location, which indicates in which hexagon the user is. Hence, with Eq. (4.9), the second factor in the first line of Eq. (4.18) is

decomposed as

$$\begin{aligned}
&\mathrm{p}\left(\boldsymbol{z}_k|\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k}, \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right) \\
&\quad = \mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k}, \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right) \qquad (4.19) \\
&\qquad \times \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}, \boldsymbol{u}_{1:k}\right) \\
&\quad = \mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{x}_{\mathrm{u},k}, \boldsymbol{M}_k\right) \\
&\qquad \times \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right). \qquad (4.20)
\end{aligned}$$

The location of the user is in general conditionally dependent on the visibility map. However, to exploit information on the visibility of transmitters at a time instant k, it is necessary to know which transmitters are visible. That knowledge is obtained from the visibility measurement vector. Thus, without explicit knowledge on the visibility measurement, the visibility map does not reveal any information about the user location. The numerator in the second line of Eq. (4.18) can be reformulated following

$$\begin{aligned}
&\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}, \boldsymbol{M}_k|\boldsymbol{x}_{\mathrm{u},k-1}, \boldsymbol{M}_{k-1}, \boldsymbol{u}_k\right) \\
&\quad = \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}|\boldsymbol{x}_{\mathrm{u},k-1}, \boldsymbol{M}_{k-1}, \boldsymbol{u}_k\right) \qquad (4.21) \\
&\qquad \times \mathrm{p}\left(\boldsymbol{M}_k|\boldsymbol{x}_{\mathrm{u},k-1:k}, \boldsymbol{M}_{k-1}, \boldsymbol{u}_k\right) \\
&\quad = \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}|\boldsymbol{x}_{\mathrm{u},k-1}, \boldsymbol{u}_k\right) \\
&\qquad \times \mathrm{p}\left(\boldsymbol{M}_k|\boldsymbol{x}_{\mathrm{u},k}, \boldsymbol{M}_{k-1}\right). \qquad (4.22)
\end{aligned}$$

In Eq. (4.22), $\boldsymbol{M}_k$ is independent from $\boldsymbol{x}_{\mathrm{u},k-1}$ if $\boldsymbol{M}_{k-1}$ is known, since the knowledge of the user state at time instant $k-1$ with respect to the visibility map is already contained in $\boldsymbol{M}_{k-1}$.

As described in Section 2.2.3, choosing the importance density is a crucial step in the design of the particle filter. One aspect is that it should be designed such that samples can be drawn from it easily. The denominator in the second line of Eq. (4.18) equals

$$\begin{aligned}
&\mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},k}, \boldsymbol{M}_k|\boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{M}_{0:k-1}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \\
&\quad = \mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},k}|\boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{M}_{0:k-1}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \qquad (4.23) \\
&\qquad \times \mathrm{q}\left(\boldsymbol{M}_k|\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k-1}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \\
&\quad = \mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},k}|\boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{M}_{0:k-1}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \\
&\qquad \times \mathrm{q}\left(\boldsymbol{M}_k|\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k-1}, \boldsymbol{z}_{\mathrm{R},1:k}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{u}_{1:k}\right). \qquad (4.24)
\end{aligned}$$

The importance density in the user-map particle filter is defined such that

$$\begin{aligned}
&\mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},k}, \boldsymbol{M}_k|\boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{M}_{0:k-1}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \\
&\quad := \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}|\boldsymbol{x}_{\mathrm{u},k-1}, \boldsymbol{u}_k\right) \mathrm{p}\left(\boldsymbol{M}_k|\boldsymbol{x}_{\mathrm{u},k}, \boldsymbol{M}_{k-1}, \boldsymbol{z}_{\mathrm{V},k}\right). \qquad (4.25)
\end{aligned}$$

The denominator in the second line of Eq. (4.18) is therefore

$$\begin{aligned}
&\mathrm{q}\left(\boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_k^{<i>}|\boldsymbol{x}_{\mathrm{u},0:k-1}^{<i>}, \boldsymbol{M}_{0:k-1}^{<i>}, \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \\
&\quad := \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}^{<i>}|\boldsymbol{x}_{\mathrm{u},k-1}^{<i>}, \boldsymbol{u}_k\right) \mathrm{p}\left(\boldsymbol{M}_k^{<i>}|\boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_{k-1}^{<i>}, \boldsymbol{z}_{\mathrm{V},k}\right). \qquad (4.26)
\end{aligned}$$

To sample a new user-map particle, the second line in Eq. (4.26) is evaluated from left to right. The first factor corresponds to the movement model of the user. With the choice of the movement model as in Eq. (3.21), a new user particle $\boldsymbol{x}_{\mathrm{u},k}^{<i>}$ can be drawn given the user particle $\boldsymbol{x}_{\mathrm{u},k-1}^{<i>}$ at time instant $k-1$ and the control input $\boldsymbol{u}_k$.

The new user particle $\boldsymbol{x}_{\mathrm{u},k}^{<i>}$ is then used in the second factor in the second line of Eq. (4.26) to draw the new visibility map $\boldsymbol{M}_k^{<i>}$ given $\boldsymbol{M}_{k-1}^{<i>}$, the user particle $\boldsymbol{x}_{\mathrm{u},k}^{<i>}$ and the measurements $\boldsymbol{z}_{\mathrm{V},k}$ of the visibilities of transmitters.

The visibility measurements are used to learn the visibility map $\boldsymbol{M}_k^{<i>}$. As the user moves through the scenario, the numbers $C_{h,k-1}^{<i,j>}$ and $\bar{C}_{h,k-1}^{<i,j>}$ count the number of times that the j^{th} transmitter has been visible and not visible, respectively, from the h^{th} hexagon by the i^{th} user-map particle up to time instant $k-1$. For sampling $\boldsymbol{M}_k^{<i>}$, the numbers $C_{h,k-1}^{<i,j>}$ and $\bar{C}_{h,k-1}^{<i,j>}$ are updated with the visibility measurement $\boldsymbol{z}_{\mathrm{V},k}$ to $C_{h,k}^{<i,j>}$ and $\bar{C}_{h,k}^{<i,j>}$ for the h^{th} hexagon for all transmitters. If the j^{th} transmitter was visible,

$$C_{h,k}^{<i,j>} = C_{h,k-1}^{<i,j>} + 1, \tag{4.27}$$

and otherwise

$$\bar{C}_{h,k}^{<i,j>} = \bar{C}_{h,k-1}^{<i,j>} + 1. \tag{4.28}$$

The user state $\boldsymbol{x}_{\mathrm{u},k}^{<i>}$ is only necessary to identify the index h of the hexagon in which the i^{th} user-map particle is located at time instant k.

A visibility measurement $z_{\mathrm{V},k}^{<j>}$ can be regarded as the realization of a random variable that follows a binomial distribution. Since the Beta distribution is the conjugate prior of the normal distribution, the parameters $p_{h,k}^{<i,j>}$ and $q_{h,k}^{<i,j>}$ of the corresponding Beta distribution are updated accordingly to

$$\begin{aligned} p_{h,k}^{<i,j>} &= C_{h,k}^{<i,j>} + \nu_h^{<j>}, \quad \text{and} \\ q_{h,k}^{<i,j>} &= \bar{C}_{h,k}^{<i,j>} + \bar{\nu}_h^{<j>}, \end{aligned} \tag{4.29}$$

respectively. In short, drawing a new visibility map is done by updating the parameters of the Beta distributions of all transmitters for the hexagon in which the corresponding user-map particle is located.

Inserting Eqs. (4.20), (4.22) and (4.26) into Eq. (4.18) yields

$$\begin{aligned} w_k^{<i>} &\propto w_{k-1}^{<i>} \, \mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k} | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_k^{<i>}\right) \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\ &\quad \times \frac{\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}^{<i>} | \boldsymbol{x}_{\mathrm{u},k-1}^{<i>}, \boldsymbol{u}_k\right) \mathrm{p}\left(\boldsymbol{M}_k^{<i>} | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_{k-1}^{<i>}\right)}{\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}^{<i>} | \boldsymbol{x}_{\mathrm{u},k-1}^{<i>}, \boldsymbol{u}_k\right) \mathrm{p}\left(\boldsymbol{M}_k^{<i>} | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_{k-1}^{<i>}, \boldsymbol{z}_{\mathrm{V},k}\right)} \\ &= w_{k-1}^{<i>} \, \mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k} | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_k^{<i>}\right) \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\ &\quad \times \frac{\mathrm{p}\left(\boldsymbol{M}_k^{<i>} | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_{k-1}^{<i>}\right)}{\mathrm{p}\left(\boldsymbol{M}_k^{<i>} | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_{k-1}^{<i>}, \boldsymbol{z}_{\mathrm{V},k}\right)}. \end{aligned} \tag{4.30}$$

With Bayes' theorem, the denominator in the last line of Eq. (4.30) can be expressed as

$$\mathrm{p}\left(\boldsymbol{M}_k|\boldsymbol{x}_{\mathrm{u},k},\boldsymbol{M}_{k-1},\boldsymbol{z}_{\mathrm{V},k}\right)=\mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{x}_{\mathrm{u},k},\boldsymbol{M}_{k-1:k}\right)\frac{\mathrm{p}\left(\boldsymbol{M}_k|\boldsymbol{x}_{\mathrm{u},k},\boldsymbol{M}_{k-1}\right)}{\mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{x}_{\mathrm{u},k},\boldsymbol{M}_{k-1}\right)}. \quad (4.31)$$

Inserting Eq. (4.31) into Eq. (4.30) leads to

$$\begin{aligned} w_k^{<i>} &\propto w_{k-1}^{<i>}\,\mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{M}_k^{<i>}\right)\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>},\boldsymbol{z}_{\mathrm{V},1:k},\boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\ &\quad\times\frac{\mathrm{p}\left(\boldsymbol{M}_k^{<i>}|\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{M}_{k-1}^{<i>}\right)}{\mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{M}_{k-1:k}^{<i>}\right)\frac{\mathrm{p}\left(\boldsymbol{M}_k^{<i>}|\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{M}_{k-1}^{<i>}\right)}{\mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{M}_{k-1}^{<i>}\right)}} \\ &= w_{k-1}^{<i>}\,\mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{M}_{k-1}^{<i>}\right)\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>},\boldsymbol{z}_{\mathrm{V},1:k},\boldsymbol{z}_{\mathrm{R},1:k-1}\right). \end{aligned} \quad (4.32)$$

The last factor in the last line of Eq. (4.32) is the likelihood with respect to the radio measurements. It can be calculated as

$$\begin{aligned} &\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>},\boldsymbol{z}_{\mathrm{V},1:k},\boldsymbol{z}_{\mathrm{R},1:k-1},\boldsymbol{u}_{1:k}\right) \\ &\quad=\prod_{j=1}^{N_{\mathrm{TX},k}^{\mathrm{a}}}\sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{k-1}^{<i,j,\ell>}\,\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{z}_{\mathrm{V},k},\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right), \end{aligned} \quad (4.33)$$

where

$$\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{z}_{\mathrm{V},k},\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right)=\begin{cases}\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right) & \text{if } z_{\mathrm{V},k}^{<j>}\neq 0\\ 1 & \text{if } z_{\mathrm{V},k}^{<j>}=0\end{cases}, \quad (4.34)$$

and the likelihood $\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right)$ is evaluated as in Eq. (3.22). The derivation of Eq. (4.33) following [Gen+16a] can be found in Appendix B.2. The visibility measurement $\boldsymbol{z}_{\mathrm{V},k}$ in the term $\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{z}_{\mathrm{V},k},\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right)$ in Eqs. (4.33) and (4.34) serves as an indicator whether the j^{th} transmitter is visible at time instant k.

The second factor in the last line of Eq. (4.32) is the likelihood with respect to the visibility measurements. The visibility measurements are dependent on $\boldsymbol{M}_{k-1}^{<i>}$ and the user-map particle. The user-map particle is only required to identify the hexagon in which the user-map particle is located at time instant k. If the index of that hexagon is h,

$$\begin{aligned} \mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{M}_{k-1}^{<i>}\right)&=\mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{M}_{h,k-1}^{<i>}\right) \\ &=\prod_{j=1}^{N_{\mathrm{TX},k}^{\mathrm{a}}}\mathrm{p}\left(z_{\mathrm{V},k}^{<j>}|\boldsymbol{M}_{h,k-1}^{<i,j>}\right), \end{aligned} \quad (4.35)$$

where it is assumed that the visibility measurements $z_{\mathrm{V},k}^{<j>}$ for the single transmitters are independent from each other. The factors in the second line of

Eq. (4.35) can be calculated as the expectation values of the corresponding Beta distributions as in Eqs. (4.7) and (4.8), namely

$$\mathrm{p}\left(z_{\mathrm{V},k}^{<j>}|\boldsymbol{M}_{h,k-1}^{<i,j>}\right) = \begin{cases} \frac{p_{h,k-1}^{<i,j>}}{p_{h,k-1}^{<i,j>}+q_{h,k-1}^{<i,j>}} & \text{if } z_{\mathrm{V},k}^{<j>} \neq 0 \\ \frac{q_{h,k-1}^{<i,j>}}{p_{h,k-1}^{<i,j>}+q_{h,k-1}^{<i,j>}} & \text{if } z_{\mathrm{V},k}^{<j>} = 0. \end{cases} \tag{4.36}$$

Inserting Eqs. (4.33) and (4.35) into Eq. (4.32) reveals the final equation for the weight update for the i^{th} user-map particle in the user-map particle filter as

$$w_k^{<i>} \propto w_{k-1}^{<i>} \prod_{j=1}^{N_{\mathrm{TX},k}^{\mathrm{a}}} \mathrm{p}\left(z_{\mathrm{V},k}^{<j>}|\boldsymbol{M}_{h,k-1}^{<i,j>}\right) \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{k-1}^{<i,j,\ell>} \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{z}_{\mathrm{V},k}, \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right). \tag{4.37}$$

The weight update in the $N_{\mathrm{TX},k}^{\mathrm{a}}$ transmitter particle filters follows directly as

$$w_k^{<i,j,\ell>} \propto \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{k-1}^{<i,j,\ell>} \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{z}_{\mathrm{V},k}, \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right). \tag{4.38}$$

The weights in Eqs. (4.37) and (4.38) are calculated only up to proportionality and need to be normalized in the respective particle filters.

The number of hexagons and the number of transmitters grow dynamically as the user moves. When the user enters a new hexagon with index h at time instant k, this hexagon is added to the visibility map, and the numbers $C_{h,k-1}^{<i,j>}$ and $\bar{C}_{h,k-1}^{<i,j>}$ are initialized with zero for all transmitters. Likewise, if KEST detects a new signal component with index j, a new transmitter is added for all hexagons. In this case, the new transmitter has never been visible in any hexagon at previous time instants. Thus, $C_{h,k-1}^{<i,j>} = 0$, and $\bar{C}_{h,k-1}^{<i,j>}$ is the number of time instants during which the i^{th} user-map particle has been located in the hexagon with index h from time instants zero to $k-1$.

The Channel-SLAM algorithm from Chapter 3 with the Rao-Blackwellized particle filter derived above is denoted by the term Channel-SLAM with visibility information, and presented in Algorithm 4 for one time instant $k > 0$. Adding and discarding transmitters are not considered in Algorithm 4.

4.2.4 Complexity

As mentioned above, the memory requirement of a visibility map estimate is $2N_H N_{\mathrm{TX}}^{\mathrm{a}}$ positive numbers, namely $C_{h,k}^{<i,j>}$ and $\bar{C}_{h,k}^{<i,j>}$, which need to be stored for every user-map particle. It grows as more transmitters are detected and more hexagons are visited.

Estimating and exploiting visibility information has negligible influence on the computational complexity of Channel-SLAM. Compared to the standard Channel-SLAM algorithm from Chapter 3, two additional calculations need to be performed for every user-map particle, namely for drawing a new visibility map and for calculating the new user-map particle weights.

Algorithm 4: Channel-SLAM with visibility information at one time instant $k > 0$.

Input:
user-map particles $[\boldsymbol{x}_{\mathrm{u},0:k-1} \;\; \boldsymbol{M}_{0:k-1}]^{<i>}$ and weights $w_{0:k-1}^{<i>}$ for $i = 1, ..., N_p$,
transmitter particles $\boldsymbol{x}_{\mathrm{u},0:k-1}^{<i,j,\ell>}$ and weights $w_{0:k-1}^{<i,j,\ell>}$ for $i = 1, ..., N_p$, $j = 1, ..., N_{\mathrm{TX},k}^{\mathrm{a}}$, $\ell = 1, ..., N_{p,\mathrm{Tx}}$,
measurements $\boldsymbol{z}_{\mathrm{R},k}$ and $\boldsymbol{z}_{\mathrm{V},k}$, and control input $\boldsymbol{u}_k$
Output:
user-map particles $[\boldsymbol{x}_{\mathrm{u},k} \;\; \boldsymbol{M}_k]^{<i>}$ and weights $w_k^{<i>}$ for $i = 1, ..., N_p$,
transmitter particles $\boldsymbol{x}_{\mathrm{u},k}^{<i,j,\ell>}$ and weights $w_k^{<i,j,\ell>}$ for $i = 1, ..., N_p$, $j = 1, ..., N_{\mathrm{TX},k}^{\mathrm{a}}$, $\ell = 1, ..., N_{p,\mathrm{Tx}}$,

for $i = 1, ..., N_p$ **do**
  draw a new user particle $\boldsymbol{x}_{\mathrm{u},k}^{<i>}$ from $\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k} | \boldsymbol{x}_{\mathrm{u},k-1}^{<i>}, \boldsymbol{u}_k\right)$;
  draw a new visibility map $\boldsymbol{M}_k^{<i>}$ from $\mathrm{p}\left(\boldsymbol{M}_k | \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_{k-1}^{<i>}, \boldsymbol{z}_{\mathrm{V},k}\right)$;
  for $j = 1, ..., N_{TX,k}$ **do**
    for $\ell = 1, ..., N_{p,\,Tx}$ **do**
      draw a new transmitter particle $\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}$ from $\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k} | \boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j,\ell>}\right)$;
      calculate the weight $w_k^{<i,j,\ell>}$ with Eq. (4.38) ;
  calculate the weight $w_k^{<i>}$ with Eq. (4.37);
normalize all user-map and transmitter weights with Eq. (2.29) ;
if *any user-map or transmitter particles suffer from degeneracy* **then**
  resample the corresponding particles

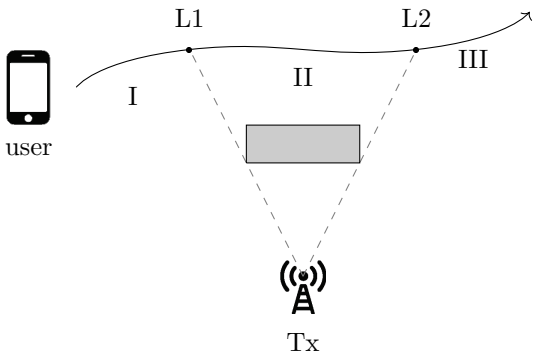


Figure 4.6: The user receives the signal from transmitter Tx in region I, loses the signal in region II due to blocking, and receives it again in region III. The user needs to associate the signal received in region III with the transmitter Tx.

Drawing a new visibility map is described in Line 3 of Algorithm 4. It reduces to updating $2N^{\mathrm{a}}_{\mathrm{TX},k}$ parameters based on the visibility measurement for the hexagon the user-map particle is located in as described in Section 4.2.3. Thus, for each user-map particle, $2N^{\mathrm{a}}_{\mathrm{TX},k}$ integer additions are performed.

The user-map particle weight update is performed in Line 6 of Algorithm 4. Compared to the standard Channel-SLAM algorithm, the additional $N^{\mathrm{a}}_{\mathrm{TX},k}$ factors in Eq. (4.35) need to be evaluated following Eq. (4.36).

4.3 Data Association

Data association is a crucial part of any long-term robust SLAM scheme. It essentially tries to answer the question which measurements stem from which landmarks. An example for the data association problem in wireless localization is depicted in Fig. 4.6. The user travels along the trajectory in three regions. In region I, the signal from the transmitter Tx is received. After passing the point L1 and entering region II, the signal is lost due to blocking by an obstacle. The signal is received again in region III after passing the point L2 and needs to be associated with the transmitter Tx.

In Channel-SLAM, data association tries to find correspondences among the transmitters. This problem is equivalent to finding which signal components correspond at different time instants, i.e., which signal components correspond to the same propagation path. In KEST, the tracking nature of the Kalman filters inherently yields associations among signal components from one time instant to another. However, once the track of a signal component is lost and detected later again as in Fig. 4.6, KEST can not associate the two signal components. Instead, KEST treats the re-detected signal component in region III as a new signal component that had not been observed before.

When KEST detects a signal component, a new transmitter is initialized based on the measurement, or KEST estimate, for this transmitter. The uncer-

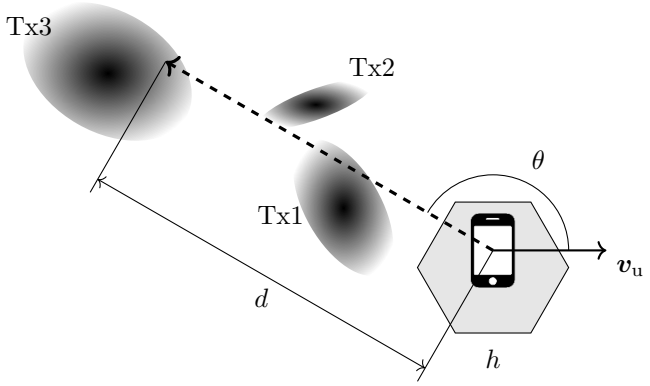


Figure 4.7: The estimated PDFs for the locations of three transmitters are exemplarily plotted by the shaded ellipses. Given the estimated ToA $\tau = {}^{d}/_{c_0}$ and AoA θ for a newly detected signal component, this new signal component may be associated with any of the three transmitters.

tainty about the transmitter's state tends to be high upon initialization. This issue is due to the fact that Channel-SLAM considers an underdetermined system at one single time instant as described in Section 3.3. Hence, if a user travels through a scenario loosing track of transmitters and detecting new or even previously observed transmitters, the uncertainty about the user state grows. However, if a new signal component is detected by KEST and a correct association of this new and a previously observed transmitter is made, the initial uncertainty about the new transmitter's state is reduced considerably. With a perfect data association, the uncertainty about the user state is bounded if a sufficient number of transmitters from a limited set of transmitters is visible throughout the user trajectory.

For notational simplicity, the transmitters that had been visible before time instant $k-1$, but are not visible currently at time instant k, are called *old* transmitters. Whenever a new signal component and thus a new transmitter is detected by KEST, the challenge of data association in Channel-SLAM is to decide between the following two cases.

1. The new transmitter is indeed a new transmitter that had never been visible before.

2. The new transmitter corresponds to an old transmitter that had been visible before.

In the second case, the question is to which of the old transmitters the new one corresponds.

The data association problem for Channel-SLAM is exemplarily depicted in Fig. 4.7. While the user is located in the h^{th} hexagon at time instant k, KEST detects a new signal component with radio measurement $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$ with ToA $\tau = {}^{d}/_{c_0}$ and AoA θ. The current location estimate PDFs for the three old transmitters

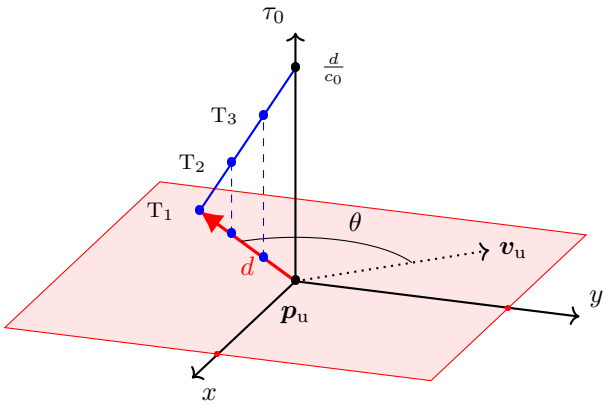


Figure 4.8: The location space is the x-y plane indicated by the red rectangle, and the delay offset space corresponds to the τ_0 axis. A ToA measurement $\tau = {}^{d}/_{c_0}$ and an AoA measurement θ may correspond to any transmitter located along the blue line.

Tx1, Tx2, and Tx3 are depicted by the ellipses. The question of data association is now whether the new signal component with radio measurement $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$ corresponds to one of the three old transmitters, and if so, to which one, or to a new one.

Due to the unknown clock offsets of the transmitters, each transmitter location on the dashed line might be suitable. This issue is further illustrated by Fig. 4.8. The location space in the x-y plane is indicated by the red rectangle. The τ_0 axis pointing up depicts the delay offset space of a transmitter. The location space and the delay offset space together form the transmitter state space. Thus, a transmitter can be represented as a point in the three-dimensional space in Fig. 4.8, as it is described by a location and a delay offset.

The user location $\boldsymbol{p}_\mathrm{u}$ is in the origin of the coordinate system for simplicity, and the user is oriented towards $\boldsymbol{v}_\mathrm{u}$. The measurement $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$ for the new transmitter consists of the ToA $\tau = {}^{d}/_{c_0}$ and the AoA θ. Both measured quantities are in the location space.

The measurement $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$ could correspond to a transmitter with a state anywhere on the blue line alike. If the location of the transmitter corresponding to $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$ is closer to the user in the location space, it has a higher clock offset. The red line labeled d represents the ToA measurement and is the projection of the blue line onto the location space. Therefore, the transmitter location could be anywhere on the red line if the delay offset is unknown. Exemplarily, three possible transmitter states T_1, T_2 and T_3 are depicted with their locations projected into the location space for the measurement $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$. I.e., each of the three states fits the measurement $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$ regarding their location in the x-y plane and their delay offset on the τ_0 axis. If the delay offset is zero, for example, the transmitter is located in the x-y plane at the point T_1.

In the absence of AoA measurements, the blue line extends to a cone around

the τ_0 axis that cuts the x-y plane in a circle with radius d around the origin. The new transmitter with measurement $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$ is then located anywhere on or within this circle, making the data association problem even harder.

Wrong association decisions can introduce a bias in the state estimate of both the user and the transmitters. In the worst case, they cause a filter to diverge. The underdetermined system at one time instant k in Channel-SLAM can cause association ambiguities and thus wrong decisions. Hence, the robustness and reliability of data association is of high importance.

In many other SLAM problems, data association is a rather easy problem, since it is easy to differentiate among the single landmarks. For example in SLAM schemes based on visual cameras, landmarks can be differentiated based on their appearance [Low+16]. In multipath assisted positioning schemes, data association is hard, since virtual transmitters arise only due different propagation paths of the very same transmit signal. The fundamental problem is the lack of diversity among the physical and virtual transmitters. The information on transmitter visibilities increases this diversity considerably.

In the following, a data association scheme for the Channel-SLAM algorithm with visibility information is derived, which is based on both the transmitter state and visibility estimates. In Section 4.3.1, it is assumed that no more than one new signal component is detected by KEST at one time instant. A generalization to the case where KEST detects more than one new signal component is described in Section 4.3.2.

4.3.1 Data Association for One New Transmitter

As mentioned above, wrong association decisions cause biases in the estimates and may even cause the filter to diverge. Hence, a multiple hypothesis tracking (MHT) association method is applied. Every user-map particle makes a hard decision on associations. The user-map particle state space is extended by association decisions. The association decisions of one particle are one hypothesis for the correct associations. The ensemble of user-map particles with their different hypotheses can be regarded as a soft decision. In the following, a data association scheme for one user-map particle is derived. For notational simplicity, the particle index i may be omitted in the association variables.

Assuming that KEST detects one new signal component, i.e., one new transmitter, at time instant k, the measurement vector for this new transmitter is $\tilde{\boldsymbol{z}}_k^{\mathrm{new}} = [\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}T} \;\; \tilde{\boldsymbol{z}}_{\mathrm{V},k}^{\mathrm{new}}]^T$. Since the new signal component has been detected and we assume only one new signal component, $\tilde{\boldsymbol{z}}_{\mathrm{V},k}^{\mathrm{new}} = 1$ is the index of the ToA measurement in $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$. The set of indices of old transmitters that have not been associated with any transmitter at time instant k is denoted by Υ_k. The set Υ_k thus contains indices of transmitters that the new transmitter can be associated with. The association variable a_k is the index of an old transmitter that the new transmitter is associated with. It can take any of the values in Υ_k or zero, in which case no association is made, and a new transmitter is initialized based on $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$. The set of previous association decisions from time instants zero to $k-1$ is denoted by $\mathcal{A}_{k-1}$. It contains tuples (a, ν), which mean that the new

transmitter with index ν has been associated with the old transmitter with index a. Consequently, the ν^{th} transmitter has been initialized with the state of the a^{th} transmitter.

First, the marginalized likelihoods for an association of the new transmitter with every transmitter with index $a_k \in \Upsilon_k$ are calculated for time instant k by [Thr+04]

$$p_{a_k} \coloneqq \mathrm{p}\left(\tilde{\boldsymbol{z}}_k^{\text{new}} | a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\text{u},k}^{<i>}, \boldsymbol{M}_k^{<i>}, \boldsymbol{z}_{1:k-1}\right) \tag{4.39}$$

$$\begin{aligned} &= \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\text{R},k}^{\text{new}} | a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\text{u},k}^{<i>}, \boldsymbol{M}_k^{<i>}, \boldsymbol{z}_{1:k-1}\right) \\ &\quad \times \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\text{V},k}^{\text{new}} | a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\text{u},k}^{<i>}, \boldsymbol{M}_k^{<i>}, \boldsymbol{z}_{1:k-1}\right), \end{aligned} \tag{4.40}$$

which are explicitly conditioned on the association variables a_k and $\mathcal{A}_{k-1}$. The first factor in Eq. (4.40) is the likelihood regarding the radio measurement, and the second factor the likelihood regarding the visibility. In the second factor, the user state is only relevant for the knowledge that the user-map particle is located in the h^{th} hexagon. The second factor is denoted by

$$\begin{aligned} p_{v,a_k} &\coloneqq \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\text{V},k}^{\text{new}} | a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\text{u},k}^{<i>}, \boldsymbol{M}_k^{<i>}, \boldsymbol{z}_{1:k-1}\right) \\ &= \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\text{V},k}^{\text{new}} | a_k, \boldsymbol{M}_{h,k}^{<i>}\right), \end{aligned} \tag{4.41}$$

and indicates the probability that the new transmitter is visible at time instant k in the h^{th} hexagon conditioned on the association with the old transmitterwith index a_k and the estimated visibility map. It is calculated as the expectation value

$$p_{v,a_k} = \mathbb{E}_{h,k}^{<i,a_k>}. \tag{4.42}$$

The first factor in Eq. (4.40) can be calculated as

$$\begin{aligned} p_{r,a_k} &\coloneqq \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\text{R},k}^{\text{new}} | a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\text{u},k}^{<i>}, \boldsymbol{M}_k^{<i>}, \boldsymbol{z}_{1:k-1}\right) \\ &= \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\text{R},k}^{\text{new}} | a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\text{u},k}^{<i>}, \boldsymbol{z}_{1:k-1}\right) \\ &= \int \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\text{R},k}^{\text{new}} | \boldsymbol{x}_{\text{TX},k}^{<i,a_k>}, a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\text{u},k}^{<i>}, \boldsymbol{z}_{\text{R},1:k-1}\right) \\ &\quad \times \mathrm{p}\left(\boldsymbol{x}_{\text{TX},k}^{<i,a_k>} | a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\text{u},k}^{<i>}, \boldsymbol{z}_{1:k-1}\right) \mathrm{d}\boldsymbol{x}_{\text{TX},k}^{<i,a_k>}, \end{aligned} \tag{4.43}$$

since the radio measurement is independent from the visibility map and measurements. The first factor in the integral of Eq. (4.43) can be simplified to

$$\begin{aligned} &\mathrm{p}\left(\tilde{\boldsymbol{z}}_{\text{R},k}^{\text{new}} | \boldsymbol{x}_{\text{TX},k}^{<i,a_k>}, a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\text{u},k}^{<i>}, \boldsymbol{z}_{\text{R},1:k-1}\right) \\ &\quad = \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\text{R},k}^{\text{new}} | \boldsymbol{x}_{\text{TX},k}^{<i,a_k>}, a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\text{u},k}^{<i>}\right). \end{aligned} \tag{4.44}$$

The second factor in the integral of Eq. (4.43) is the PDF of the ${a_k}^{\text{th}}$ transmitter after the prediction and before the update in the particle filter. Thus,

the second factor equals

$$\begin{aligned} &\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k>}|a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\ &\quad = \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{k-1}^{<i,a_k,\ell>} \delta\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k>} - \boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k,\ell>-}\right), \end{aligned} \tag{4.45}$$

where $\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k,\ell>-}$ is the ℓ^{th} particle of the ${a_k}^{\mathrm{th}}$ transmitter after the prediction, but before the update in the corresponding particle filter at time instant k. Therefore, its associated weight $w_{k-1}^{<i,a_k,\ell>}$ is the weight at time instant $k-1$.

Substituting Eq. (4.44) and Eq. (4.45) into Eq. (4.43) yields the radio measurement likelihoods

$$p_{r,a_k} = \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{k-1}^{<i,a_k,\ell>} \, \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}|\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k,\ell>-}, a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\mathrm{u},k}^{<i>}\right), \tag{4.46}$$

which can be evaluated analogue to Eq. (3.22).

Finally,

$$p_{a_k} = p_{r,a_k} p_{v,a_k} \tag{4.47}$$

with Eqs. (4.46) and (4.42), respectively.

For complexity reasons, only transmitters with indices a_k for which p_{a_k} exceeds a threshold ρ are regarded for associations. The set of indices of such old transmitters at time instant k is denoted by

$$\Gamma_k = \{j : j \in \Upsilon_k \wedge p_j > \rho\}. \tag{4.48}$$

The probability p_0 denotes the probability that no association is made. Accordingly, if $a_k = 0$ is chosen as association, no association is made and the new transmitter's state is initialized based on the radio measurement. In the following, two principles to come to an association decision (a, ν) between the new transmitter with index ν and the old transmitter with index a for the i^{th} user-map particle are considered.

1. With the ML method, the association for which the likelihood is maximized is chosen deterministically, i.e.,

$$\hat{a}_{\mathrm{ML},k}^{<i>} = \underset{j \in \Gamma_k \cup \{0\}}{\arg\max}\, p_j. \tag{4.49}$$

2. With the data association sampling (DAS) method, an association $\hat{a}_{\mathrm{DAS},k}^{<i>}$ is sampled randomly from the set $\Gamma_k \cup \{0\}$ based on the corresponding likelihoods from Eq. (4.47).

With data association sampling, association decisions are expected to be spread wider across the range of possible associations. On the one hand, this increases the computational complexity, since more user-map particles are required. On the other hand, allowing associations that seem less likely at one time instant k improves the robustness of data association.

As mentioned above, wrong associations cause in general biases in the estimates or even the particle filter to diverge. However, the weights of particles with wrong association decisions are very likely to considerably decrease over time, since the measurement likelihoods in Eqs. (4.33) and (4.36) are likely to become small. Hence, such particles are likely to not be resampled at future time instants. In this way, the user-map particle filter inherently removes wrong association decisions over time.

One user-map particle holds now the hypothesis of the user state vector $\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}$, the transmitter visibility map $\boldsymbol{M}_{0:k}^{<i>}$, the PDFs of the transmitters' states $\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{1:k}\right)$ for $j = 1, \ldots, N_{\mathrm{TX},k}^{\mathrm{a}}$, and the set of data association decisions $\mathcal{A}_k^{<i>}$. The information regarding one user-map particle is represented by

$$\left\langle \left[\boldsymbol{x}_{\mathrm{u},0:k} \;\; \boldsymbol{M}_{0:k}\right]^{<i>}, \mathcal{A}_k^{<i>}, \left\{ \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{R},1:k}\right)\right\}_{j=1,\ldots,N_{\mathrm{TX},k}^{\mathrm{a}}} \right\rangle. \quad (4.50)$$

4.3.2 Data Association for Multiple New Transmitters

The above association method can be applied to find an association between a new and an old transmitter if only one new transmitter is initialized at a time instant k. This constraint is dropped in the following, as KEST may detect multiple new signal components at a single time instant with estimates

$$\tilde{\boldsymbol{z}}_k^{\mathrm{new}} = [\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}T} \;\; \tilde{\boldsymbol{z}}_{\mathrm{V},k}^{\mathrm{new}T}]^T. \quad (4.51)$$

The vector $\tilde{\boldsymbol{z}}_{\mathrm{V},k}^{\mathrm{new}}$ contains the indices of the ToA estimates in $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}}$ for the $N_{\mathrm{TX},k}^{\mathrm{new}}$ new transmitters at time instant k and is thus simply the vector

$$\tilde{\boldsymbol{z}}_{\mathrm{V},k}^{\mathrm{new}} = \begin{bmatrix} 1 & 2 & \ldots & N_{\mathrm{TX},k}^{\mathrm{new}} \end{bmatrix}^T. \quad (4.52)$$

The set of the $N_{\mathrm{TX},k}^{\mathrm{new}}$ new transmitter indices that are initialized at time instant k for the new signal components is denoted by $\mathcal{V}_k$. The association between the old transmitter with index $a \in \Upsilon_k$ and the new transmitter with index $\nu \in \mathcal{V}_k$ is represented by the tuple (a, ν). As in Section 4.3.1, an association decision is made by every user-map particle individually. Nevertheless, the particle index i and the time instant index k may be omitted in the following.

Analogue to Section 4.3.1, the marginalized likelihoods for associations can be calculated as

$$p_{(a,\nu)} = p_{r,(a,\nu)} p_{v,(a,\nu)}, \quad (4.53)$$

where

$$p_{r,(a,\nu)} = \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{k-1}^{<i,a,\ell>} \, \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}<\nu>}|\boldsymbol{x}_{\mathrm{TX},k}^{<i,a,\ell>}, a, \mathcal{A}_{k-1}, \boldsymbol{x}_{\mathrm{u},k}^{<i>}\right) \quad (4.54)$$

is the radio measurement likelihood regarding the radio measurement $\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}<\nu>}$ for the ν^{th} newly initialized transmitter similar to Eq. (4.46). Analogue to Eq. (4.41),

$$p_{v,(a,\nu)} = \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\mathrm{V},k}^{\mathrm{new}<\nu>}|a, \boldsymbol{M}_{h,k}^{<i>}\right) = \mathbb{E}_{h,k}^{<i,a>} \quad (4.55)$$

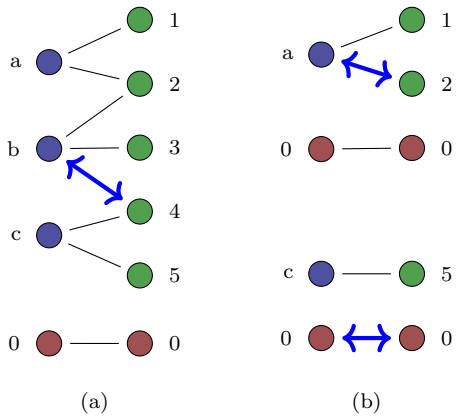


Figure 4.9: The greedy algorithm for associating multiple new signal components represented in purple on the left hand side of the bipartite graphs with previously observed transmitters represented by the green circles on the right hand side of the graphs works recursively. After the association decision indicated by the blue double headed arrow in (a), the involved nodes b and 4 are deleted from the graph, and the algorithm is applied on the remaining subgraphs in (b). In each graph, the edge between the two red nodes represents the decision for no association being made.

indicates the probability that the new transmitter with index a is visible at time instant k in the h^{th} hexagon, in which the corresponding user-map particle is located, conditioned on the association with the old transmitter with index a and the visibility map $\boldsymbol{M}_{h,k}^{<i>}$.

Again, only associations with marginalized likelihoods exceeding the threshold ρ are regarded. The set of possible associations is thus

$$\Gamma_k = \left\{(a,\nu) : a \in \Upsilon_k \wedge \nu \in \mathcal{V}_k \wedge p_{(a,\nu)} > \rho\right\}. \tag{4.56}$$

Association decisions are again based on a ML method or on data association sampling. In the ML method, the association

$$(\widehat{a,\nu})_{\text{ML},k}^{<i>} = \underset{(a,\nu)\in\Gamma_k\cup\left\{(0,0)\right\}}{\arg\max} p_{(a,\nu)} \tag{4.57}$$

is chosen, where the association tuple $(0,0)$ with likelihood $p_{(0,0)}$ represents the case where no association is made. With data association sampling, an association $(\widehat{a,\nu})_{\text{DAS},k}^{<i>}$ is sampled based on the probabilities $p_{(a,\nu)}$.

For choosing associations from the set Γ_k, a greedy algorithm is applied, which is exemplarily illustrated in Fig. 4.9. First, a bipartite graph is constructed with the newly initialized transmitters with indices in $\mathcal{V}_k$ on the one side, and the old transmitters with indices in Υ_k on the other side. For each possible association $(a,\nu) \in \Gamma_k$, an edge is drawn connecting the two corresponding

nodes if $p_{(a,\nu)} > \rho$. In Fig. 4.9 (a), $\mathcal{V}_k = \{a, b, c\}$ with nodes in blue on the left, and $\Upsilon_k = \{1, 2, 3, 4, 5\}$ with nodes in green on the right, and the edges correspond to the possible associations in Γ_k. One additional edge is drawn between the red nodes. This edge represents the decision for no association with association tuple $(0, 0)$.

Each edge in the graph represents a possible association. The edge between the nodes with indices $a \in \mathcal{V}_k$ and $\nu \in \Upsilon_k$ representing the association with tuple $(a, \nu) \in \Gamma_k$ is weighted by $p_{(a,\nu)}$. With the ML method or data association sampling, a decision is made for one association $(\widehat{a,\nu})$ from Γ_k based on the likelihoods $p_{(a,\nu)}$. While the ML method decides for the edge with highest weight in the graph as in Eq. (4.57), the data association sampling method samples one edge randomly based on the weights of the edges. The edge weights thus determine the order in which transmitters are chosen for associations.

If $(\widehat{a,\nu}) = (0, 0)$, the algorithm returns and no further associations decisions are made. Otherwise, $(\widehat{a,\nu}) \neq (0, 0)$, and the set of association decisions $\mathcal{A}_k$ is updated with the new association $(\widehat{a,\nu})$. The association tuples that involve the associated transmitters a and ν are removed from Γ_k, since each transmitter can be associated only once. In Fig. 4.9 (a), an association between the old transmitter with index 4 and the new transmitter with index b is made, indicated by the blue double-headed arrow. The corresponding edge and all nodes the edge is connected to are removed from the graph. The above procedure is then repeated for all remaining subgraphs considering the tuple $(0, 0)$ representing no association, indicated by the two red nodes in each graph in Fig. 4.9. In Fig. 4.9 (b), the two remaining subgraphs are drawn. In the upper subgraph, the association decision for the old transmitter with index 2 with the new transmitter with index a is made. In the lower subgraph, the decision for no further association, i.e, the tuple $(0, 0)$, is made. Hence, the tuples $(4, b)$ and $(2, a)$ are added to $\mathcal{A}_{k-1}$ to obtain $\mathcal{A}_k$. The new transmitters a and b are initialized with the state estimate of the old transmitters 2 and 4, respectively. The new transmitter with index c is newly initialized based on the corresponding measurement.

The greedy algorithm based on the graph is summarized for one user-map particle i in Algorithm 5. The output of the algorithm is a list of tuples representing associations, which are used to update $\mathcal{A}_{k-1}$ to obtain $\mathcal{A}_k$. In particular, the output of the algorithm is a hard decision of associations. As mentioned above, since association decisions are made by each user-map particle individually, the ensemble of association decisions regarding all user-map particles can be regarded as a soft decision output.

4.3.3 Complexity

The main computational complexity in the data association scheme explained in Section 4.3.2 is the calculation of the radio likelihoods in Eq. (4.54). The number of these likelihoods is $|\Upsilon_k| N_{\mathrm{TX},k}^{\mathrm{new}}$, corresponding to all possible associations between an old transmitter from Υ_k and a new transmitter from $\mathcal{V}_k$. The number of visibility likelihoods that need to be evaluated as in Eq. (4.55) is the number of old transmitters $|\Upsilon_k|$.

Algorithm 5: Greedy algorithm for data association

Input:
new transmitter indices $\mathcal{V}_k$ and old transmitters indices Υ_k,
user-map particle $[\boldsymbol{x}_{\mathrm{u},k} \;\; \boldsymbol{M}_k]^{<i>}$,
transmitter states $\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}$ for $j \in \Upsilon_k$,
measurements $\tilde{\boldsymbol{z}}_k^{\mathrm{new}<\nu>}$ for $\nu \in \mathcal{V}_k$
Output:
list of associations

create the set Γ_k from Υ_k and $\mathcal{V}_k$ as in Eq. (4.56);
create the bipartite graph as in Fig. 4.9 from Γ_k;
decide for an association $(\widehat{a,\nu})$ with ML or DAS from $\Gamma_k \cup (0,0)$;
if *decision for no association, i.e.,* $(\widehat{a,\nu}) = (0,0)$*,* **then**
    return;
else
    add association $(\widehat{a,\nu})$ to list of associations;
    remove associated nodes from bipartite graph;
    apply this algorithm on all remaining subgraphs with the corresponding subsets of $\mathcal{V}_k$ and Υ_k;

For the greedy algorithm in Algorithm 5, the computational worst case is that $\min\{|\Upsilon_k|, N_{\mathrm{TX},k}^{\mathrm{new}}\}$ associations are made. Thus, the maximum number of recursions of the greedy algorithm is $\min\{|\Upsilon_k|, N_{\mathrm{TX},k}^{\mathrm{new}}\}$.

The number of times that the data association algorithm is performed during the user movement is bounded by the overall number of transmitters a user detects minus one, i.e., $N_{\mathrm{TX},k}^{\mathrm{a}} - 1$. In this worst case, there is never more than one transmitter visible at any time instant, i.e., $N_{\mathrm{TX},k} \leq 1$ for all k. Hence, the $N_{\mathrm{TX},k}^{\mathrm{a}}$ detected transmitters are visible one after another. Only for the first transmitter, the data association scheme can not be applied since at that time, no old transmitters are mapped by the user. For the $N_{\mathrm{TX},k}^{\mathrm{a}} - 1$ other transmitters, the data association scheme is applied.

The evaluation of the factors in Eq. (4.47) contributes negligibly to the overall computational complexity of Channel-SLAM. However, the state space of the user is increased by the association decisions. A higher dimension of the state space in a particle filter requires more particles for the same robustness and positioning performance. The increase of user-map particles due to the data association scheme can significantly increase the computational complexity. Nevertheless, the robustness of the data association scheme against false associations limits the amount of additional particles needed. In the best case, all data associations are correct and no additional particles need to be deployed at all. Since the data association scheme incorporates visibility information about transmitters, it turns out to be very robust.

4.4 Summary and Outlook

This chapter has introduced the concept of visibility of transmitters. The new particle filter that has been derived estimates visibility maps of transmitters jointly with the user state and the states of the transmitters. The estimated visibility maps can improve the positioning performance of Channel-SLAM if a location that a user had been visited before is visited again.

While the diversity among transmitters is by nature poor in multipath assisted positioning, it is drastically increased if information regarding the visibility of the transmitters is incorporated. Data association in SLAM is heavily dependent on diversity among landmarks. Thus, a data association scheme was derived for Channel-SLAM that incorporates information regarding the visibility of transmitters for an increased association robustness.

Most important, the diversity gained by estimating the visibility of transmitters is a key enabler for user cooperation in Channel-SLAM. We will see in the next chapter how the increased diversity allows users to cooperate in terms of sharing transmitter maps for an improved positioning performance.

Chapter 5

Cooperative Channel-SLAM

In scenarios such as malls, museums or public buildings, many users travel through the same scenario on different trajectories. Users can cooperate by exchanging information on transmitters. The first user in the scenario creates a transmitter map with Channel-SLAM, and hands the map on. The second user exploits the knowledge in the transmitter map obtained from the first user. Since the second user takes radio and visibility measurements from different positions, the original map can be improved with this new information on the states and visibilities of the transmitters in the map. In terms of transmitter states, the uncertainty tends to shrink when measurements from different users at different positions are made. In terms of visibilities, the information becomes more and more reliable and extensive. Furthermore, the second user may add information on transmitters that the first user has never observed to the map. Finally, the second user hands the updated map over to the third user. Users can in this way iteratively use and improve transmitter maps in a crowdsourcing scheme [SG11]. The shared transmitter map is continuously updated with information from new users.

A transmitter map can be handed on from one user to another directly, or managed by a local unit. The local unit could be the physical transmitter in a scenario, for example. If cellular signals are used, it is a base station, and if WLAN signals are used, it is a WLAN router.

Section 5.1 shows how to create a comprehensive transmitter map from the transmitters' states and visibilities estimated with Channel-SLAM. In Section 5.2, a scheme to share and use the information within transmitter maps among users is presented. A method to merge maps from different users is shown in Section 5.3.

5.1 Creating Maps

In the Rao-Blackwellized particle filter for Channel-SLAM with visibility information derived in Section 4.2, each user-map particle represents a weighted hypothesis of the user state, a transmitter state map, a transmitter visibility map and a set of association decisions as in Eq. (4.50).

To create a *comprehensive transmitter map* that can be shared with other users, the information from all N_p user-map particles' transmitter maps is merged. The set of indices of user-map particles that have not associated the j^{th} transmitter with any old transmitter is denoted by $\mathcal{T}_j$. It is formally defined as

$$\mathcal{T}_j = \left\{ i : \nexists\, (j, \cdot) \in \mathcal{A}^{<i>} \right\}, \tag{5.1}$$

where $(j, \cdot)$ is a tuple representing an association between the old transmitter with index j with any new transmitter. Thus, if $i \in \mathcal{T}_j$, there is no tuple of the form $(j, \cdot)$ in the set of association decisions $\mathcal{A}^{<i>}$ of the i^{th} user-map particle, where $\cdot$ may be the index of any new transmitter. Otherwise, if the i^{th} user-map particle has decided for an association between the j^{th} transmitter with a new transmitter at some point, there is a tuple $(j, \cdot)$ in $\mathcal{A}^{<i>}$, and $i \notin \mathcal{T}_j$.

A transmitter state map in Channel-SLAM consists of a set of posterior PDFs of transmitter states, which are represented by particle clouds. In the comprehensive transmitter state map, the particle cloud representing the state $\boldsymbol{x}_{\text{TX},k}^{<\cdot,j>}$ of the j^{th} transmitter is calculated by marginalizing over the user-map particles as

$$\mathrm{p}\left(\boldsymbol{x}_{\text{TX},k}^{<\cdot,j>} | \boldsymbol{z}_{1:k}, \boldsymbol{x}_{\text{u},0:k}\right) = \sum_{i \in \mathcal{T}_j} w_k^{<i>} \sum_{\ell=1}^{N_{p,\text{Tx}}} w_k^{<i,j,\ell>} \delta\left(\boldsymbol{x}_{\text{TX},k}^{<\cdot,j>} - \boldsymbol{x}_{\text{TX},k}^{<i,j,\ell>}\right). \tag{5.2}$$

After applying Eq. (5.2), the number of particles in the PDF for a transmitter is very high. The particle reduction method from [Gen+17b] mentioned in Section 3.6 is applied to decrease the memory occupation of the state map. The PDF in Eq. (5.2) regarding the j^{th} transmitter is reformulated as

$$\mathrm{p}\left(\boldsymbol{x}_{\text{TX}}^{<\cdot,j>}\right) = \sum_{\ell=1}^{N_{p,\text{Tx}}^*} w^{<\cdot,j,\ell>} \delta\left(\boldsymbol{x}_{\text{TX}}^{<\cdot,j>} - \boldsymbol{x}_{\text{TX}}^{<\cdot,j,\ell>}\right), \tag{5.3}$$

where $\boldsymbol{x}_{\text{TX}}^{<\cdot,j,\ell>}$ is the ℓ^{th} transmitter particle and $w^{<\cdot,j,\ell>}$ its associated weight, and the time instant index k is left out. The number of particles in the particle cloud marginalized over the user-map particles is denoted by $N_{p,\text{Tx}}^*$. The weights need to be normalized to one. An alternative representation of a transmitter particle with the clock offset interpreted as an additional propagation distance is denoted by

$$\tilde{\boldsymbol{x}}_{\text{TX}}^{<\cdot,j,\ell>} = \boldsymbol{T}_{c_0} \boldsymbol{x}_{\text{TX}}^{<\cdot,j,\ell>} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & c_0 \end{bmatrix} \boldsymbol{x}_{\text{TX}}^{<\cdot,j,\ell>}. \tag{5.4}$$

To obtain a comprehensive visibility map, the counters of the visibilities of the j^{th} transmitter in the h^{th} hexagon are proportionally added up to marginalize over the user-map particles as

$$\begin{aligned} C_h^{<\cdot,j>} &= \sum_{i \in \mathcal{T}_j} w_k^{<i>} C_{h,k}^{<i,j>}, \quad \text{and} \\ \bar{C}_h^{<\cdot,j>} &= \sum_{i \in \mathcal{T}_j} w_k^{<i>} \bar{C}_{h,k}^{<i,j>}. \end{aligned} \tag{5.5}$$

With these counters, the parameters of the Beta distributions for the j^{th} transmitter in the h^{th} hexagon in the comprehensive visibility map are

$$\begin{aligned} p_h^{<j>} &= C_h^{<\cdot,j>} + \nu_h^{<j>}, \quad \text{and} \\ q_h^{<j>} &= \bar{C}_h^{<\cdot,j>} + \bar{\nu}_h^{<j>}. \end{aligned} \tag{5.6}$$

Due to different association decisions, there is a certain belief in that a transmitter in the comprehensive map exists in the first place. For example, if all user-map particles have decided for an association between a transmitter with index j with any other transmitter when the transmitter with index j was initialized, the belief that this transmitter exists is zero. In general, this belief in the existence of the j^{th} transmitter is denoted by $\mathrm{p}_{e,\mathrm{u}}^{<j>}$ and calculated as

$$\mathrm{p}_{e,\mathrm{u}}^{<j>} = \sum_{i \in \mathcal{T}_j} w_k^{<i>}. \tag{5.7}$$

In addition to the belief in the existence of a transmitter, a measure of the reliability of the transmitter is defined in the following. In particular, the *reliability distance* of a transmitter denotes the distance that the user has traveled while the transmitter was visible. The reliability distance of the j^{th} transmitter is defined as

$$\mathrm{d}_{\mathrm{rel,u}}^{<j>} = \sum_{i \in \mathcal{T}_j} w_k^{<i>} \mathrm{d}_{\mathrm{rel,u}}^{<i,j>}, \tag{5.8}$$

where $\mathrm{d}_{\mathrm{rel,u}}^{<i,j>}$ is the distance traveled by the i^{th} user-map particle while the j^{th} transmitter was visible. For example, if the j^{th} transmitter was only visible between time instants three and five, but not at any other time instant, then

$$\mathrm{d}_{\mathrm{rel,u}}^{<i,j>} = \|\boldsymbol{p}_{\mathrm{u},4}^{<i>} - \boldsymbol{p}_{\mathrm{u},3}^{<i>}\| + \|\boldsymbol{p}_{\mathrm{u},5}^{<i>} - \boldsymbol{p}_{\mathrm{u},4}^{<i>}\|, \tag{5.9}$$

where $\boldsymbol{p}_{\mathrm{u},k}^{<i>}$ denotes the location of the i^{th} user-map particle at time instant k.

The reasoning behind the reliability distance is that the uncertainty in a transmitter's state estimate tends to shrink as measurements from different user positions are made. Hence, with a long distance covered by the user while the transmitter is visible, the uncertainty about the transmitter's state tends to decrease. In contrast, a transmitter's state estimate tends to be uncertain if its reliability distance is short.

Finally, a comprehensive transmitter map consists of a set of transmitters, where each transmitter is characterized by

1. a PDF denoting the transmitter's state, calculated by Eq. (5.2),
2. a hexagonal visibility map, where the parameters of the Beta distributions are obtained from Eqs. (5.5) and (5.6),
3. a probability of existence as in Eq. (5.7), and
4. a reliability distance, given by Eq. (5.8).

As mentioned above, comprehensive transmitter maps may be shared among users. Transmitters that are located far away from the user trajectories tend to have a high uncertainty in their state estimate due to a bad geometrical dilution of precision. On the one hand, many particles are needed to represent their state PDF, which considerably increases the memory occupation of a transmitter map. On the other hand, they do not contribute a lot to the positioning of a user due to the uncertainty about their state. Likewise, transmitters that have been visible only for a short time tend to have a high uncertainty in their state estimate since Channel-SLAM considers an underdetermined problem at one time instant as stated in Section 3.5.

When maps of transmitters are exchanged among users, transmitters with a high uncertainty about their state are therefore removed from the map. The mean and the covariance matrix of the PDF of the j^{th} transmitter's state estimate in the alternative representation as in Eq. (5.4) are

$$\tilde{\boldsymbol{\mu}}_{\text{TX}}^{<j>} = \sum_{\ell=1}^{N_{p,\text{Tx}}^{*}} w^{<\cdot,j,\ell>} \tilde{\boldsymbol{x}}_{\text{TX}}^{<\cdot,j,\ell>} \tag{5.10}$$

and

$$\tilde{\boldsymbol{C}}_{\text{TX}}^{<j>} = \sum_{\ell=1}^{N_{p,\text{Tx}}^{*}} w^{<\cdot,j,\ell>} \left(\tilde{\boldsymbol{x}}_{\text{TX}}^{<\cdot,j,\ell>} - \tilde{\boldsymbol{\mu}}_{\text{TX}}^{<j>}\right)\left(\tilde{\boldsymbol{x}}_{\text{TX}}^{<\cdot,j,\ell>} - \tilde{\boldsymbol{\mu}}_{\text{TX}}^{<j>}\right)^{T}, \tag{5.11}$$

respectively. The j^{th} transmitter is removed from the map if the trace of the covariance matrix in the alternative representation exceeds a threshold τ_σ, i.e., if

$$\text{Tr}\left(\tilde{\boldsymbol{C}}_{\text{TX}}^{<j>}\right) > \tau_\sigma. \tag{5.12}$$

To further decrease the memory occupation, transmitters that are unlikely to exist due to data association decisions are removed from the comprehensive transmitter map. If the probability of existence of the j^{th} transmitter as in Eq. (5.7) is below a threshold τ_{ex}, it is removed from the map.

5.2 Exchanging Maps

Channel-SLAM is a relative positioning system in the sense that a user estimates their own location relative to the transmitters in the environment. The user coordinate system is chosen arbitrarily, and in general no relation to a global

coordinate system is known. Likewise, when a user receives a comprehensive transmitter map from a different user or some central entity, such a map is in a coordinate system different from the user's.

The transmitter map estimated by a user with Channel-SLAM is denoted by the term user map. A comprehensive transmitter map that a user receives from a third party and that is used as prior knowledge is denoted by the term prior map. The two maps are represented in different coordinate systems that are related by a two-dimensional translation $\bar{\boldsymbol{\gamma}} = [\bar{x} \;\; \bar{y}]^T$ and a rotation $\bar{\varphi}$. In order to be able to exploit the information in a prior map, these three unknown transformation parameters need to be estimated. In addition to the transformation parameters, correspondences among transmitters in the user map and the prior map need to be found. A transmitter in the user map may or may not have a correspondence to a transmitter in the prior map, and vice versa. A set of such correspondences of transmitters in the two maps is denoted by $\mathcal{C}$. It contains tuples of the form $(j_\mathrm{u}, j_\mathrm{p})$, where j_u is the index of a transmitter in the user map and j_p the index of the corresponding transmitter in the prior map. The number of corresponding transmitters and thus the cardinality of the set $\mathcal{C}$ is denoted by $N_\mathcal{C}$.

Estimating both the transformation parameters and the correspondences among transmitters is denoted by the term *map matching*. The algorithm to match two maps is explained in the following sections.

5.2.1 The Map Matching Algorithm

An overview of the map matching algorithm is illustrated in Fig. 5.1. The inputs are the user and the prior map. The first stage is a modified random sample consensus (RANSAC) algorithm framed by the dashed line. The idea of RANSAC is presented in Section 5.2.2. The modified RANSAC adapted to the map matching problem consists of four blocks that are framed by the dotted line and repeated N_it times. First, a number of transmitters from each of the two maps is chosen randomly, which is discussed in Section 5.2.3. The corresponding block is labeled R1. With the choice of the transmitters, the correspondences among the transmitters are estimated in the block labeled R2, which is described in Section 5.2.4. The transformation parameters are estimated in the block R3, which is explained in detail in Section 5.2.5. Afterward, a local optimization scheme is applied to increase the robustness and accuracy of the parameter estimates in the block labeled R4. The local optimization scheme is discussed in Section 5.2.6

For each of the N_it iterations, one solution, or hypothesis, of a map match is obtained. The s^th such hypothesis is denoted by H_s. It is a set comprising the estimated transformation parameters $\bar{\boldsymbol{\gamma}}_s$ and $\bar{\varphi}_s$ and the set of correspondences $\mathcal{C}_s$ of size $N_{\mathcal{C},s}$. A set of correspondences is also denoted by the terms correspondence set or consensus set. Each hypothesis H_s obtained from the optimization in the RANSAC algorithm is added to the set of hypotheses $\mathcal{H}$.

The above method provides hypotheses for map matching that are based on the transmitter state estimates in the two maps. The actual decision for the best hypothesis $\hat{H}$ is made in a likelihood ratio test based on the visibility

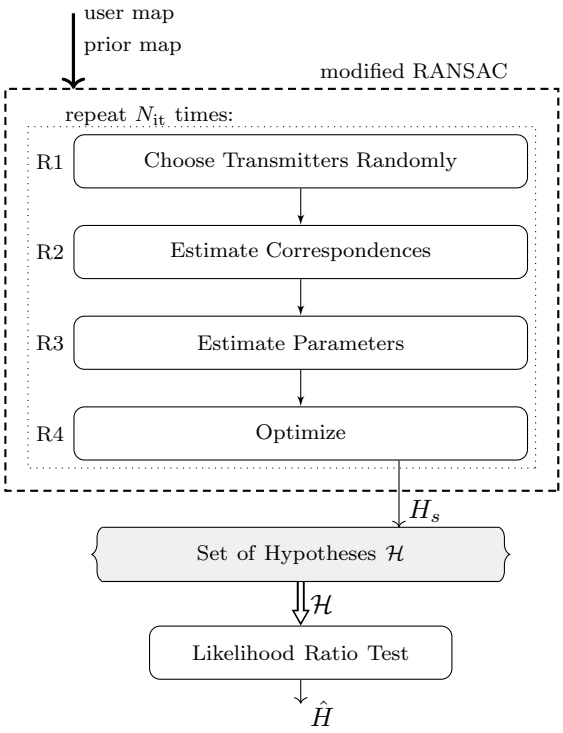


Figure 5.1: Based on the user and the prior map, map matching works in two stages as indicated by the flow chart. First, a modified RANSAC algorithm outputs a number of hypotheses of map matches based on the transmitter states. A likelihood ratio test based on the visibility information about transmitters chooses the best hypothesis.

estimates of the transmitters in the maps. This likelihood ratio test is explained in detail in Section 5.2.7.

5.2.2 The RANSAC Algorithm

The RANSAC algorithm [FB81] is a tool to estimate the parameters of a mathematical model given a noisy data set that is corrupted by outliers. Outliers are data points that do not match the model. In contrast, inliers are data points supporting the model. RANSAC has been used mainly in computer vision for image stitching [Sze06]. There is a variety of modifications and extensions to the original RANSAC algorithm in the literature [RFP08; HNM13].

A generic RANSAC algorithm is summarized in Algorithm 6. The inputs are the data set, also called the complete set, and the mathematical model. RANSAC returns the estimated parameters for the model, after testing a number of hypotheses for model parameters in an iterative fashion. The number of iterations is often chosen such that a correct result can be found with a certain probability. In each iteration, a set of data points is chosen randomly from the data set. In the original RANSAC algorithm, the cardinality of this set, i.e., the number of data points that are chosen, is the minimum number of data points that are required to uniquely calculate the model parameters. With the chosen data points, the model parameters are calculated. Afterward, the solution is evaluated by testing each data point from the complete set against the model. Typically, the decision whether a data point supports the model is taken on a threshold basis. The data points supporting the model are inliers, and they are defined as the consensus set. The hypothesis is then added to the set $\mathcal{H}$.

Finally, the best of the hypotheses in $\mathcal{H}$ is chosen and the corresponding model parameters are returned. Optimality often refers to the cardinality of the correspondence set, since a solution supporting many data points is considered very reliable. In this case, the parameters that are supported by the most data points in the complete set are returned. An important feature of RANSAC is that it inherently estimates which data points are inliers and which are outliers.

Algorithm 6: A generic RANSAC algorithm

Input: a data set and a mathematical model
Output: parameters of the mathematical model

```
for a certain number of times do
    choose a random set of data points from the data set;
    calculate the model parameters for these data points;
    find the consensus set;
    add the hypothesis to the set H;
find the best hypothesis set from H and return the corresponding model
 parameters;
```

5.2.3 Number of Iterations

In RANSAC, N_T data points are selected randomly at each iteration as in Line 2 of Algorithm 6. Typically, this number N_T is chosen to be the minimum number of data points that is required to calculate a unique solution for the parameter set that needs to be estimated. In map matching, there are three unknown transformation parameters, and thus $N_T = 3$ transmitters in the user map and $N_T = 3$ transmitters in the prior map are chosen randomly in each iteration of the modified RANSAC in the block labeled R1 in Fig. 5.1. Each iteration corresponds to a RANSAC hypothesis, and each hypothesis assumes that all $N_T = 3$ data points that were chosen are inliers. I.e, it is assumed that each of the N_T chosen transmitters in the user map has a correspondence in the N_T transmitters chosen from prior map.

The number N_{it} of iterations that are performed in RANSAC in Line 1 of Algorithm 6 is determined in a probabilistic manner. It is defined such that the probability that a correct solution for the transmitter correspondences is chosen in at least one iteration exceeds a threshold p_{succ}. Choosing a correct solution means choosing transmitters in the user map that actually do correspond to the transmitters chosen from the prior map. In the following considerations, it is assumed that there are at least N_T inliers in the two maps.

Denote the probability of sampling N_T of the $N_{\mathcal{U}}$ transmitters in the user map and the corresponding N_T of the $N_{\mathcal{P}}$ transmitters in the prior map by q. This is the probability that in one iteration of RANSAC, N_T inliers from the user map and their corresponding transmitters in the prior map are sampled in arbitrary order. Since q depends on the actual number of inliers N_{IN}, it is expressed explicitly as a function $\mathrm{q}\left(N_{\mathrm{IN}}\right)$.

In at least one of the N_{it} RANSAC iterations, a set of N_T corresponding transmitters in each map shall be sampled with a probability that is not smaller than p_{succ}. If N_{it} iterations of RANSAC are performed, the probability that a set of only inliers has not been sampled shall be smaller than $1-p_{\mathrm{succ}}$. Therefore,

$$\left(1-\mathrm{q}\left(N_{\mathrm{IN}}\right)\right)^{N_{\mathrm{it}}} \leq 1-p_{\mathrm{succ}}. \tag{5.13}$$

Rearranging Eq. (5.13) yields that the number of iterations required is

$$N_{\mathrm{it}} \geq \left\lceil \frac{\log\left(1-p_{\mathrm{succ}}\right)}{\log\left(1-\mathrm{q}\left(N_{\mathrm{IN}}\right)\right)} \right\rceil. \tag{5.14}$$

The probability to sample N_T inliers out of the $N_{\mathcal{U}}$ transmitters in the user map is

$$\mathrm{q_u}\left(N_{\mathrm{IN}}\right) = \frac{\binom{N_{\mathrm{IN}}}{N_T}}{\binom{N_{\mathcal{U}}}{N_T}}. \tag{5.15}$$

The probability to sample the corresponding N_T transmitters in the prior map in arbitrary order is

$$\mathrm{q_a} = \frac{1}{\binom{N_{\mathcal{P}}}{N_T}}. \tag{5.16}$$

Finally,

$$\mathrm{q}\left(N_{\mathrm{IN}}\right) = \mathrm{q_u}\left(N_{\mathrm{IN}}\right)\mathrm{q_a} = \frac{\binom{N_{\mathrm{IN}}}{N_T}}{\binom{N_{\mathcal{U}}}{N_T}\binom{N_{\mathcal{P}}}{N_T}}. \tag{5.17}$$

The number N_{IN} of inliers is usually unknown. Nevertheless, since $\mathrm{q}\left(N_{\mathrm{IN}}\right)$ is a function that increases monotonically with N_{IN}, N_{IN} can be conservatively estimated to be the cardinality of the largest set of inliers found so far. It is initialized with $N_{\mathrm{IN}} = N_T = 3$. Whenever a larger consensus set is found, the estimate of N_{IN} is updated. The number of iterations is thus adapted on the fly as RANSAC is operating.

To calculate N_{it}, it is assumed that given N_T transmitters from the user map and the corresponding transmitters in the prior map, the actual correspondences are estimated perfectly. A scheme to estimate the correspondences is presented in Section 5.2.4.

5.2.4 Correspondence Estimation

Given the N_T randomly chosen transmitters from each map, the goal is now to find the best set of correspondences for these transmitters in the block labeled R2 in Fig. 5.1. The set of the N_T chosen transmitter indices in the user map is denoted by $\mathcal{U}$, while the set of the N_T chosen transmitter indices in the prior map is $\mathcal{P}$.

For notational convenience, a tuple in a correspondence set is denoted within this section by $(\mathtt{U}_q, \mathtt{P}_r)$, which describes a correspondence between a transmitter with index $\mathtt{U}_q \in \mathcal{U}$ from the user map and a transmitter with index $\mathtt{P}_r \in \mathcal{P}$ from the prior map.

Since there are $N_T!$ possibilities to arrange N_T transmitters, there are also $N_{\mathcal{C}} = N_T!$ distinct possible sets of correspondences of N_T transmitters for each RANSAC hypothesis. In the s^{th} RANSAC hypothesis, these sets are denoted by $\mathcal{C}_{s,1}, \ldots, \mathcal{C}_{s,N_{\mathcal{C}}}$. For $N_T = 3$, the number of possible correspondence sets is $N_{\mathcal{C}} = 6$.

Because the transformation parameters between the coordinate systems of the user and the prior map are unknown, the absolute locations of transmitters in the two maps do not contain any relevant information. However, the relative distances among transmitters in each map are used to obtain the set of correspondences $\hat{\mathcal{C}}_s$.

The shape of a transmitter's state PDF depends on the user's trajectory and the corresponding geometrical dilution of precision for that transmitter. This shape can thus be very different for different user trajectories as exemplarily shown in Fig. 5.2. The particles of the location estimates of three different transmitters are shown after the user has traveled on two different user trajectories that are drawn in blue in Fig. 5.2 (a) and (b). The start and end positions of the user are labeled START and END, respectively. In the scenario, there is one physical transmitter Tx1 indicated by the red upward triangle. The transmit signal is reflected at the wall drawn in black and scattered at the point scatterer. The corresponding virtual transmitters are vTx2 and vTx3. The location of vTx3 coincides with the location of the scatterer.

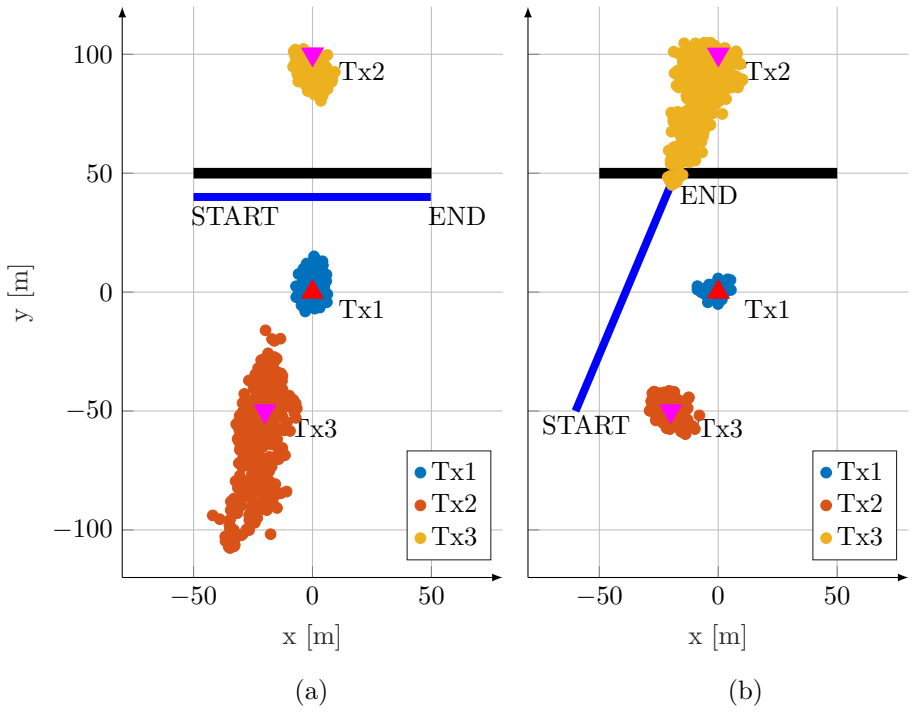


Figure 5.2: In the scenario with one physical transmitter Tx1 and two virtual transmitters Tx2 and Tx3, different user tracks in (a) and (b) lead to different estimates for estimated PDFs of the transmitter states in Channel-SLAM. The estimates are indicated by the particles in different colors. The geometrical dilution of precision influences the uncertainty in the transmitter state estimates.

While the uncertainty about the location of transmitter Tx1 is low in both cases, the uncertainty about the location of transmitter Tx2 is low in Fig. 5.2 (a) and high in Fig. 5.2 (b). For the virtual transmitter Tx3, the situation is reversed. The high or low uncertainty can be explained with a bad or good geometrical dilution of precision as in Section 2.1.3. From a transmitter point of view, the geometrical dilution of precision is low if the user takes measurements from many different locations distributed around the transmitter location. For example, in Fig. 5.2 (b), the user travels towards Tx2 from one single direction, leading to a high geometrical dilution of precision and a high uncertainty about the transmitter's state. In contrast, in Fig. 5.2 (a), the uncertainty about the state of Tx2 is considerably smaller due to the different user trajectory.

As a measure for the distance between two transmitters, standard metrics or divergences between two PDFs such as the Kullback–Leibler divergence may therefore end up in misleading results [Cal18]. Since only transmitters with a small variance as in Eq. (5.12) are stored in transmitter maps, the distance between two transmitters is defined as the Euclidean distance between the means of the two transmitters' state estimates in the alternative representation with additional propagation distances as in Eq. (5.4). It is calculated for two transmitters with indices A and B as

$$d_{\mathrm{A,B}} = \left\| \sum_{\ell=1}^{N_p^{<\mathrm{A}>}} w^{<\cdot,\mathrm{A},\ell>} \tilde{\boldsymbol{x}}_{\mathrm{TX}}^{<\cdot,\mathrm{A},\ell>} - \sum_{\ell=1}^{N_p^{<\mathrm{B}>}} w^{<\cdot,\mathrm{B},\ell>} \tilde{\boldsymbol{x}}_{\mathrm{TX}}^{<\cdot,\mathrm{B},\ell>} \right\|, \tag{5.18}$$

where $\tilde{\boldsymbol{x}}_{\mathrm{TX}}^{<\cdot,j,\ell>}$ is the ℓ^{th} of the $N_p^{<j>}$ particles of the j^{th} transmitter in the alternative representation, and $w^{<\cdot,j,\ell>}$ its associated weight.

For notational convenience, the distance $d_{\mathtt{U}_q,\mathtt{U}_{\tilde{q}}}$ between two transmitters in the user map with indices $\mathtt{U}_q$ and $\mathtt{U}_{\tilde{q}}$ is denoted by $d^{\mathtt{U}}_{q,\tilde{q}}$. Likewise, the distance $d_{\mathtt{P}_r,\mathtt{P}_{\tilde{r}}}$ of two transmitters in the prior map with indices $\mathtt{P}_r$ and $\mathtt{P}_{\tilde{r}}$ is denoted by $d^{\mathtt{P}}_{r,\tilde{r}}$.

For each of the $N_{\mathcal{C}}$ possible sets of correspondences, each distance between two transmitters in the prior map is compared to the distance between the two corresponding transmitters in the user map. The squared differences of these distances are summed up. The set of correspondences $\hat{\mathcal{C}}_s$ for which the summed squared differences are minimal is chosen. Mathematically,

$$\hat{\mathcal{C}}_s = \underset{\substack{\mathcal{C}_u \\ u=1,\dots,N_{\mathcal{C}}}}{\arg\min} \sum_{(\mathtt{U}_q,\mathtt{P}_r)\in\mathcal{C}_u \wedge (\mathtt{U}_{\tilde{q}},\mathtt{P}_{\tilde{r}})\in\mathcal{C}_u} \left(d^{\mathtt{P}}_{r,\tilde{r}} - d^{\mathtt{U}}_{q,\tilde{q}}\right)^2, \tag{5.19}$$

leading to the best solution in a least squares sense as the squared differences of the relative distances among transmitters are minimized.

Fig. 5.3 illustrates finding correspondences among transmitters in two maps. The four squares in light and dark blue on the left side with labels $\mathtt{U}_1, ..., \mathtt{U}_4$ represent the locations of four transmitters from the user map. The light and dark green circles labeled $\mathtt{P}_1, ..., \mathtt{P}_5$ are the locations of five transmitters from the prior map. The indices of transmitters that are chosen randomly in this RANSAC iteration are $\mathcal{U} = \{\mathtt{U}_2, \mathtt{U}_3, \mathtt{U}_4\}$ from the user map and $\mathcal{P} = \{\mathtt{P}_1, \mathtt{P}_2, \mathtt{P}_4\}$

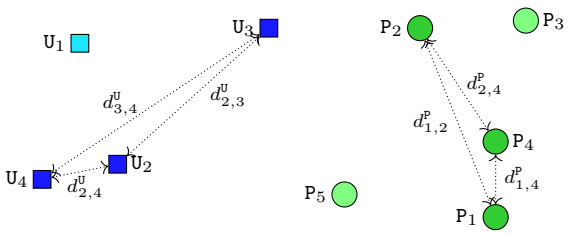


Figure 5.3: The blue squares and green circles represent locations of transmitters from a user and a prior map, respectively. The relative distances among transmitters within each map are used to obtain correspondences among transmitters in the user map with transmitters in the prior map.

from the prior map. The best set of correspondences $\hat{\mathcal{C}}_s$ is chosen as in Eq. (5.19). In Fig. 5.3, the best set is $\hat{\mathcal{C}}_s = \{(\mathtt{U}_2, \mathtt{P}_4), (\mathtt{U}_3, \mathtt{P}_2), (\mathtt{U}_4, \mathtt{P}_1)\}$.

5.2.5 Parameter Estimation

With the best correspondence set $\hat{\mathcal{C}}_s$ as obtained in Section 5.2.4, the transformation parameters are estimated in the block labeled R3 in Fig. 5.1. To simplify the notation, the transmitters in the prior map and the user map are arranged such that in $\hat{\mathcal{C}}_s$ there is a correspondence between the j^{th} transmitter in the user map and the j^{th} transmitter in the prior map for $j = 1, \ldots, N_T$. The mean of the j^{th} transmitter's state estimate in the user map is denoted by $\boldsymbol{\mu}_j^{\text{u}} = \begin{bmatrix} x_j^{\text{u}} & y_j^{\text{u}} & \tau_j^{\text{u}} \end{bmatrix}^T$. Likewise, the mean of the j^{th} transmitter's state estimate in the prior map is denoted by $\boldsymbol{\mu}_j^{\text{a}} = \begin{bmatrix} x_j^{\text{p}} & y_j^{\text{p}} & \tau_j^{\text{p}} \end{bmatrix}^T$. The means are calculated analogue to Eq. (5.10).

The means of the states of the N_T transmitters chosen from the user map are stacked in the matrix

$$\boldsymbol{L}_{\text{u}} = \begin{bmatrix} \boldsymbol{\mu}_1^{\text{u}} & \ldots & \boldsymbol{\mu}_{N_T}^{\text{u}} \end{bmatrix}^T. \tag{5.20}$$

The means of the states of the corresponding transmitters in the prior map are stacked in the same way in the matrix

$$\boldsymbol{L}_{\text{p}} = \begin{bmatrix} \boldsymbol{\mu}_1^{\text{a}} & \ldots & \boldsymbol{\mu}_{N_T}^{\text{a}} \end{bmatrix}^T. \tag{5.21}$$

With $\boldsymbol{\gamma} = [\bar{x} \;\; \bar{y} \;\; 0]$, $\boldsymbol{J} = [1 \;\; \ldots \;\; 1]^T$ of dimension $N_T \times 1$ and the rotation matrix

$$\boldsymbol{R}_{\bar{\varphi}} = \begin{bmatrix} \cos\bar{\varphi} & \sin\bar{\varphi} & 0 \\ -\sin\bar{\varphi} & \cos\bar{\varphi} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{5.22}$$

the two matrices $\boldsymbol{L}_{\text{u}}$ and $\boldsymbol{L}_{\text{p}}$ are related by

$$\boldsymbol{L}_{\text{u}} = \boldsymbol{L}_{\text{p}} \boldsymbol{R}_{\bar{\varphi}} + \boldsymbol{J}\boldsymbol{\gamma} + \boldsymbol{E}, \tag{5.23}$$

where $\boldsymbol{E}$ is a residual matrix.

The common covariance matrix $\boldsymbol{C}$ of the user and prior map transmitters is of diagonal form and defined by

$$\boldsymbol{C} = \begin{bmatrix} c_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & c_{N_T} \end{bmatrix}, \tag{5.24}$$

where $c_j = \sigma^2_{\mathrm{u},j} + \sigma^2_{\mathrm{p},j}$. The variances $\sigma^2_{\mathrm{u},j}$ and $\sigma^2_{\mathrm{p},j}$ are the traces of the covariance matrices of the j^{th} transmitter in the user map and in the prior map, respectively. These covariance matrices are calculated following Eq. (5.11) based on the corresponding transmitter particles and weights. Rearranging Eq. (5.23) yields the residual matrix

$$\boldsymbol{E} = \boldsymbol{L}_{\mathrm{u}} - \boldsymbol{L}_{\mathrm{p}}\boldsymbol{R}_{\bar{\varphi}} - \boldsymbol{J}\boldsymbol{\gamma}. \tag{5.25}$$

The in a least squares sense best transformation parameters $\hat{\bar{x}}$, $\hat{\bar{y}}$ and $\hat{\bar{\varphi}}$ are obtained by minimizing $\mathrm{Tr}\left(\boldsymbol{E}\boldsymbol{C}^{-1}\boldsymbol{E}\right)$, which is the trace of the matrix $\boldsymbol{E}\boldsymbol{C}^{-1}\boldsymbol{E}$. To obtain the minimum, $\mathrm{Tr}\left(\boldsymbol{E}\boldsymbol{C}^{-1}\boldsymbol{E}\right)$ is derived by each transformation parameter and the results are set to zero. Inserting the resulting equations into each other yields

$$\hat{\bar{\varphi}} = \arctan\left(-\frac{\sum\limits_{j=1}^{N_T} \frac{x_j^{\mathrm{u}} y_j^{\mathrm{p}}}{c_j} y_j^{\mathrm{u}} - \sum\limits_{j=1}^{N_T} \frac{x_j^{\mathrm{p}} y_j^{\mathrm{u}}}{c_j} - \frac{1}{T_C} \sum\limits_{j=1}^{N_T} \frac{x_j^{\mathrm{u}}}{c_j} \sum\limits_{j=1}^{N_T} \frac{y_j^{\mathrm{p}}}{c_j} + \frac{1}{T_C} \sum\limits_{j=1}^{N_T} \frac{y_j^{\mathrm{u}}}{c_j} \sum\limits_{j=1}^{N_T} \frac{x_j^{\mathrm{p}}}{c_j}}{\sum\limits_{j=1}^{N_T} \frac{x_j^{\mathrm{p}} x_j^{\mathrm{u}}}{c_j} + \sum\limits_{j=1}^{N_T} \frac{y_j^{\mathrm{p}} y_j^{\mathrm{u}}}{c_j} - \frac{1}{T_C} \sum\limits_{j=1}^{N_T} \frac{x_j^{\mathrm{u}}}{c_j} \sum\limits_{j=1}^{N_T} \frac{x_j^{\mathrm{p}}}{c_j} - \frac{1}{T_C} \sum\limits_{j=1}^{N_T} \frac{y_j^{\mathrm{u}}}{c_j} \sum\limits_{j=1}^{N_T} \frac{y_j^{\mathrm{p}}}{c_j}} \right), \tag{5.26}$$

where

$$T_C = \mathrm{Tr}\left(\boldsymbol{C}^{-1}\right) = \sum_{j=1}^{N_T} \frac{1}{c_j}. \tag{5.27}$$

With $\hat{\bar{\varphi}}$, the translation parameters are obtained as

$$\hat{\bar{x}} = \frac{1}{T_C} \sum_{j=1}^{N_T} \frac{x_j^{\mathrm{u}}}{c_j} - \cos\left(\hat{\bar{\varphi}}\right) \frac{1}{T_C} \sum_{j=1}^{N_T} \frac{x_j^{\mathrm{p}}}{c_j} + \sin\left(\hat{\bar{\varphi}}\right) \frac{1}{T_C} \sum_{j=1}^{N_T} \frac{y_j^{\mathrm{p}}}{c_j} \tag{5.28}$$

and

$$\hat{\bar{y}} = \frac{1}{T_C} \sum_{j=1}^{N_T} \frac{y_j^{\mathrm{u}}}{c_j} - \cos\left(\hat{\bar{\varphi}}\right) \frac{1}{T_C} \sum_{j=1}^{N_T} \frac{y_j^{\mathrm{p}}}{c_j} - \sin\left(\hat{\bar{\varphi}}\right) \frac{1}{T_C} \sum_{j=1}^{N_T} \frac{x_j^{\mathrm{p}}}{c_j}. \tag{5.29}$$

The arctan function in Eq. (5.26) returns an angle in the interval $\left]-\frac{\pi}{2}, +\frac{\pi}{2}\right[$. Hence, there is an ambiguity between the angles $\hat{\bar{\varphi}}$ and $\hat{\bar{\varphi}} + \pi$. The parameters for which $\mathrm{Tr}\left(\boldsymbol{E}\boldsymbol{C}^{-1}\boldsymbol{E}\right)$ is smaller are chosen.

The transformation is performed in the location space of the transmitters as can be seen in the form of the rotation matrix in Eq. (5.22). The clock offset of

a transmitter does also have an influence on the estimate of the transformation parameters as its uncertainty is taken into account in the common covariance matrix in Eq. (5.24).

With the transformation parameters, the prior map can be transformed into the user coordinate system. The transformed mean of the j^{th} transmitter in the prior map is

$$ {}^{\mathtt{T}}\boldsymbol{\mu}_j^{\mathtt{a}} = \left(\boldsymbol{\mu}_j^{\mathtt{a}T} \boldsymbol{R}_{\hat{\varphi}} + \hat{\boldsymbol{\gamma}} \right)^T , \tag{5.30} $$

where $\hat{\boldsymbol{\gamma}} = \begin{bmatrix} \hat{x} & \hat{y} & 0 \end{bmatrix}$.

The estimation of the transmitter correspondences in Section 5.2.4 and of the transformation parameters above corresponds to Line 3 in Algorithm 6.

In the RANSAC algorithm, the set of correspondences $\hat{\mathcal{C}}_s$ in Eq. (5.19) is expanded by all additional correspondences $(\mathtt{U}_q, \mathtt{P}_r)$ that are in accordance with the transformation parameters as in Line 4 of Algorithm 6. These additional correspondences are transmitter pairs for which the distance between the means $\boldsymbol{\mu}_{\mathtt{U}_q}^{\mathtt{u}}$ and ${}^{\mathtt{T}}\boldsymbol{\mu}_{\mathtt{P}_r}^{\mathtt{a}}$ of a transmitter $\mathtt{U}_q$ from the user map and a transformed transmitter $\mathtt{P}_r$ from the prior map, respectively, is smaller than the threshold δ_d, i.e., for which

$$ \| \boldsymbol{\mu}_{\mathtt{U}_q}^{\mathtt{u}} - {}^{\mathtt{T}}\boldsymbol{\mu}_{\mathtt{P}_r}^{\mathtt{a}} \| < \delta_d . \tag{5.31} $$

If the assignment of transmitters in the transformed prior map to transmitters in the user map is ambiguous, the Hungarian algorithm [Sch03] is used to obtain the best solution for correspondences. Such ambiguities arise if the distance from two transmitters in the prior map after the transformation to one or more transmitters in the user map is smaller than δ_d, for example. The Hungarian algorithm is a combinatorial optimization algorithm which finds the best assignment for correspondences based on the distances among transmitters.

5.2.6 Local Optimization

One assumption of the RANSAC algorithm is that a model estimated only with inliers is consistent with all other inliers from the complete data. In map matching, though, this assumption does not hold, since the data points, which are the means of transmitters in the prior and the user map, may be biased. One reason for such a bias stems from the fact that only the mean of a transmitter state estimate is regarded, while the shape of the corresponding PDF is neglected. Another reason is a possible bias from the channel estimator KEST for the signal components' parameter estimates, which translates into a bias in the transmitter state estimates.

To cope with biased estimates, a local optimization scheme as proposed in [CMK03] is applied after the transformation parameters and the consensus set are calculated in one iteration of RANSAC. This optimization scheme is performed in the block labeled R4 in Fig. 5.1. Based on the estimated transformation parameters, all inliers, i.e., transmitters from the prior map and user map that support the model, are obtained and added to the consensus set as described in Section 5.2.5. In the local optimization, the transformation parameters are re-estimated based on the consensus set, and if further correspondences

are found in accordance with Eq. (5.31), the consensus set is updated as well. The re-estimation of the parameters and the consensus set is performed repeatedly until the consensus set does not grow any further.

With the local optimization, biases in the means of the transmitter state estimates are considered. The more inliers are found, the more reliable is a solution.

5.2.7 Likelihood Ratio Test

The RANSAC algorithm described above considers only the state estimates of transmitters in the two maps. In [UG19], for example, the best of the RANSAC hypotheses is chosen as the one minimizing the error

$$E_{H,s} = \frac{1}{N_{\mathcal{C},s}} \sum_{(j_\mathrm{u}, j_\mathrm{p}) \in \mathcal{C}_s} \| \boldsymbol{\mu}_{j_\mathrm{u}}^\mathrm{u} - {}^\mathrm{T}\boldsymbol{\mu}_{j_\mathrm{p}}^\mathrm{a} \| - N_{\mathcal{C},s} \rho_\mathrm{rew}, \tag{5.32}$$

where ρ_rew is a reward term that accommodates for the fact that hypotheses with large consensus sets are considered more reliable, and $N_{\mathcal{C},s}$ is the cardinality of the consensus set of the s^th hypothesis.

While map matching with RANSAC turns out to be very robust if there are many transmitter correspondences in the two maps, i.e., if the cardinality of the actual correspondence set $\mathcal{C}$ is large, there occur map matching ambiguities in some scenarios and in the case of small actual correspondence sets. For example, if the scenario is such that the transmitter locations show certain symmetries, the correspondence estimates in Section 5.2.4 might be wrong, leading to biased estimates for the transformation parameters. As a consequence, if such a transmitter map is used by a user, the map might not bring any benefit or even bias the user's and transmitters' state estimate.

Fig. 5.4 illustrates map matching ambiguities. In Fig. 5.4 (a), the blue squares represent transmitter locations from a user map, and the green circles transmitter locations from a prior map. The transformation parameters relating the coordinate systems of the two maps are unknown. In Fig. 5.4 (b), a map match has been found with the correspondence set $\mathcal{C}_1 = \{(1, A), (2, B), (4, D)\}$ of size three. The blue dashed circles around the prior map transmitters are of radius δ_d. If the green prior map transmitters lie within such a circle, two corresponding transmitters are found. In Fig. 5.4 (c) and (d), different map matches with correspondence sets $\mathcal{C}_2 = \{(2, A), (4, D)\}$ of size two and $\mathcal{C}_3 = \{(1, B), (2, A), (3, D)\}$ of size three were found.

A fundamental problem leading to map matching ambiguities as described above is the lack of diversity among virtual transmitters, making it hard to differentiate among them. However, with visibility maps, the diversity among transmitters is increased. A transmitter is not only described by its state, i.e., location and clock offset, but also by the information from where it is visible. Including visibility information, the best RANSAC hypothesis $\hat{H}$ is chosen by

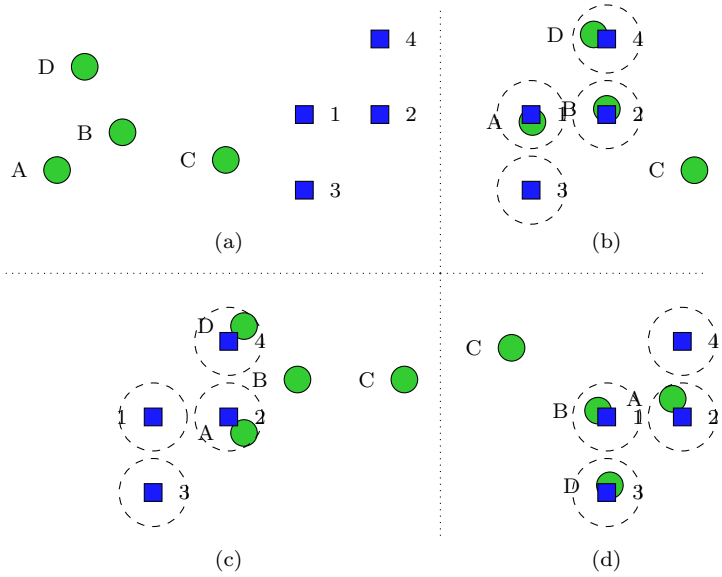


Figure 5.4: The blue squares and green circles in (a) represent locations of transmitters from the user map and from a prior map, respectively. In (b), (c) and (d), different map matching solutions are found.

a likelihood ratio test [Bav09] regarding the visibility maps, namely by

$$\begin{aligned}\hat{H} &= \underset{H_s, s=1,\ldots,N_H}{\arg\max}\ \mathrm{p}\left(H_s|\boldsymbol{M}_k\right) \\ &= \underset{H_s, s=1,\ldots,N_H}{\arg\max}\ \mathrm{p}\left(\boldsymbol{M}_k|H_s\right)\mathrm{p}\left(H_s\right),\end{aligned} \tag{5.33}$$

where $\boldsymbol{M}_k$ is the user visibility map. The prior knowledge $\mathrm{p}\left(H_s\right)$ on the s^{th} hypothesis is calculated by

$$\mathrm{p}\left(H_s\right) = \frac{\sum_{\tilde{s}=1}^{N_H} E^+_{H,\tilde{s}}}{E^+_{H,s}}, \tag{5.34}$$

where $E^+_{H,s} = \max\{E_{H,s}, 1\,\mathrm{m}\}$, and $E_{H,s}$ is obtained from Eq. (5.32). The reasoning for the max function in $E^+_{H,s}$ is that negative values in $E_{H,s}$ can falsify the results of the likelihood ratio test.

The best hypothesis in Eq. (5.33) is chosen by replacing the prior knowledge $\mathrm{p}\left(H_s\right)$ by the term in Eq. (5.34),

$$\hat{H} = \underset{H_s, s=1,\ldots,N_H}{\arg\max}\ \frac{\mathrm{p}\left(\boldsymbol{M}_k|H_s\right)}{E^+_{H,s}}. \tag{5.35}$$

To calculate the numerator in the argument of Eq. (5.35), the prior visibility map is transformed for every hypothesis H_s with the corresponding transfor-

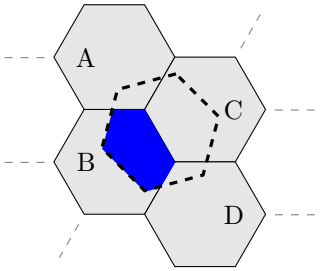


Figure 5.5: The hexagons drawn in gray are from the equivalent prior map, and thus in the user hexagonal grid. The hexagon drawn with dashed lines is a hexagon from the prior map after a translation and rotation. The visibility information from that hexagon is transferred into the equivalent prior map hexagons proportionally to the respective overlapping areas. For example, the overlapping area of the hexagon from the user map with hexagon B is drawn blue.

mation parameters into the hexagonal grid of the user's visibility map. The resulting visibility map is denoted by the term equivalent prior visibility map.

To create the equivalent prior visibility map, each hexagon in the prior visibility map is considered separately. First, the hexagon is rotated and translated according to the transformation parameters. In general, it is then not in the hexagonal grid of the user visibility map any more. Such a situation is depicted in Fig. 5.5. The hexagon with dashed edges is a hexagon from the prior map transformed with the corresponding parameters. The gray hexagons with indices A, B, C and D are hexagons in the equivalent prior map, and therefore in the hexagonal grid of the user map. The parameters of the Beta distribution for the j^{th} transmitter in the equivalent prior map are initialized with $p_B^{<\cdot,j>} = \nu_h^{<j>}$ and $q_B^{<\cdot,j>} = \bar{\nu}_h^{<j>}$ for the hexagon with index B, for example.

All hexagons in the equivalent prior map that overlap with any transformed hexagon from the prior map are found. In Fig. 5.5, the gray hexagons with indices A, B, C and D are such hexagons that overlap with the transformed hexagon. The parameters of the Beta distributions of each of the overlapping hexagons in the equivalent prior map are updated with the parameters from the transformed prior visibility map hexagon for all transmitters under consideration of the corresponding overlapping area. The overlapping area can be calculated with standard clipping algorithms such as the Sutherland–Hodgman algorithm [SH74], for example.

For example, in Fig. 5.5, the overlapping area of the transformed prior visibility map hexagon with hexagon B in the equivalent prior visibility map is a fraction $r_{B,h}^{<j>}$ of an entire hexagon's area, and it is drawn in dark blue. Thus, the Beta distribution parameters $p_B^{<\cdot,j>}$ and $q_B^{<\cdot,j>}$ in the equivalent prior visibility map hexagon with index B for the j^{th} transmitter are updated with the corresponding parameters ${}^{\text{T}}p_h^{<\cdot,j>}$ and ${}^{\text{T}}q_h^{<\cdot,j>}$ from the transformed prior map

hexagon with index h by

$$\begin{aligned} p_B^{<\cdot,j>} &= p_B^{<\cdot,j>} + r_{B,h}^{<j>\mathsf{T}} p_h^{<\cdot,j>} \quad \text{and} \\ q_B^{<\cdot,j>} &= q_B^{<\cdot,j>} + r_{B,h}^{<j>\mathsf{T}} q_h^{<\cdot,j>}. \end{aligned} \tag{5.36}$$

All hexagons in the prior visibility map are processed in the same way. The equivalent prior visibility map finally holds the information from the prior visibility map in the hexagonal grid of the user visibility map.

With the equivalent prior map, the numerator in the argument of Eq. (5.35) can be written as

$$\begin{aligned} \mathrm{p}\left(\boldsymbol{M}_k|H_s\right) = \prod_{(j_\mathrm{u},j_\mathrm{p})\in\mathcal{C}_s} \prod_{h\in\mathsf{H}_{o,s}} & \frac{1}{\mathrm{B}\left(p_{h,k}^{<j_\mathrm{u}>}, q_{h,k}^{<j_\mathrm{u}>}\right)} \\ & \times \mathbb{E}_h^{<j_\mathrm{p}>(p_{h,k}^{<j_\mathrm{u}>}-1)} (1-\mathbb{E}_h^{<j_\mathrm{p}>})^{(q_{h,k}^{<j_\mathrm{u}>}-1)}, \end{aligned} \tag{5.37}$$

where $\mathsf{H}_{o,s}$ is the set of indices of hexagons that are in both the user and the equivalent prior visibility map, $\mathbb{E}_h^{<j_\mathrm{p}>}$ is the expectation value of the Beta distribution regarding the visibility of the ${j_\mathrm{p}}^{\mathrm{th}}$ transmitter in the h^{th} hexagon in the equivalent prior map, and $p_{h,k}^{<j_\mathrm{u}>}$ and $q_{h,k}^{<j_\mathrm{u}>}$ are the parameters of the Beta distribution for the ${j_\mathrm{u}}^{\mathrm{th}}$ transmitter in the h^{th} hexagon of the user map. The expectation value of a Beta distribution is obtained as in Eq. (4.7).

For numerical stability, the terms in Eq. (5.37) and Eq. (5.35) are calculated in logarithmic domain.

The choice for the best hypothesis with the likelihood ratio test corresponds to Line 4 in Algorithm 6.

5.2.8 Summary and Complexity of Map Matching

The computational complexity of map matching is the sum of the complexity of the modified RANSAC and the complexity of the likelihood ratio test, as can be seen in Fig. 5.1.

The number of iterations in RANSAC depends on the definition of p_succ and on the number of inliers, i.e., transmitters in the two maps that actually correspond to each other. It can be calculated with Eq. (5.14).

Previous to the RANSAC iterations, the distances among transmitters in each map are calculated with Eq. (5.18). In sum, there are

$$\sum_{j=1}^{N_\mathcal{U}-1} j + \sum_{j=1}^{N_\mathcal{P}-1} j = \frac{N_\mathcal{U}(N_\mathcal{U}-1)}{2} + \frac{N_\mathcal{P}(N_\mathcal{P}-1)}{2} \tag{5.38}$$

such distances.

In each RANSAC iteration, the following steps are performed.

- Correspondence estimation as in Section 5.2.4:
  With the distances from Eq. (5.18), Eq. (5.19) is evaluated for all $N_\mathcal{C} =$

$N_T! = 6$ possible sets of correspondences to obtain the best correspondence set $\hat{\mathcal{C}}_s$. If the summed squared distances for $\hat{\mathcal{C}}_s$ in Eq. (5.19) are large, this RANSAC hypothesis is very unlikely. To decrease the computational complexity, the corresponding RANSAC iteration is aborted and the hypothesis discarded if these summed squared distances are above a threshold τ_{dist}. In this case, RANSAC continues with the next iteration.

- Parameter estimation as in Section 5.2.5:
  For the set $\hat{\mathcal{C}}$, the three transformation parameters are calculated analytically with Eqs. (5.26), (5.28) and (5.29). Further correspondences are calculated with Eq. (5.31), which needs to be evaluated $(N_{\mathcal{U}} - N_T)(N_{\mathcal{P}} - N_T)$ times for all transmitter pairs in the prior and the user map that are not in the correspondence set.

- Local optimization as in Section 5.2.6:
  The local optimization scheme tries to find additional inliers and refines the transformation parameters. The maximum number of iterations in the local optimization is $\min\{N_{\mathcal{U}} - N_T, N_{\mathcal{P}} - N_T\}$. In each iteration, the transformation parameters are calculated and new correspondences are found.

In general, the more inliers are found, the less iterations of RANSAC need to be performed. In the best case regarding the computational complexity, all transmitters in the user map have a correspondence in the prior map and vice versa, and these correspondences are found in the first iteration of RANSAC.

To reduce the computational complexity, only map matching hypotheses from RANSAC whose error as in Eq. (5.32) falls below a threshold τ_E are further considered in the likelihood ratio test. Thus, Eq. (5.32) is evaluated for all hypotheses. If no hypothesis matches the threshold, no map match was found.

In the likelihood ratio test, the following two steps are performed for each remaining hypothesis.

- First, the equivalent prior map is calculated. Each hexagon in the prior map is transformed. If it overlaps with any hexagon from the user visibility map after the transformation, its information is added proportionally to the overlapping hexagons in the user hexagon grid.

- Second, the best hypothesis is chosen with Eq. (5.35). For each hypothesis, the PDFs of the Beta distributions are evaluated in Eq. (5.37). The number of these PDFs is the size of the respective correspondence set multiplied with the number $\mathsf{H}_{o,s}$ of hexagons in the user map and the equivalent prior map that overlap.

Overall, map matching is a computationally complex process if many transmitters and hexagons are involved. In the beginning of a user track, only few transmitters have been visible to the user, and the user has come across only few hexagons. Over time, the number of visible transmitters and crossed hexagons increases.

However, since map matching is not directly a part of the actual SLAM problem in Channel-SLAM, it can be run in parallel to Channel-SLAM. Furthermore, the state and visibility estimates of the user change only little from one time instant to another. Thus, map matching does not need to be performed at every single time instant, as the map matching results are likely to be very similar across short time periods. Instead, map matching is performed every time when the user has covered a certain distance d_{int}, or when the uncertainty in the estimates of the state of one or more transmitters has shrunk by a certain extent. In this way, map matching does not increase the computational complexity of Channel-SLAM significantly.

Once a map match has been found, the prior map is transformed into the coordinate system of the user. For the transmitter states, each transmitter particle is transformed according to the estimated transformation parameters analogue to Eq. (5.30). For the visibility map, the corresponding equivalent visibility map obtained in Section 5.2.7 can be taken over directly. The transmitters in the transformed prior map can then be used for data association by the user.

5.2.9 Data Association with Prior Maps

When KEST detects a new signal component, the corresponding new transmitter is associated with an old transmitter that had been observed before by the user, with a transmitter from the prior map, or with no transmitter.

In accordance with Section 4.3.1, the set of indices of transmitters in the prior map that have not yet been associated at time instant k is denoted by Υ_k^p, and the index of the transmitter in the prior map that a new transmitter is associated with by a_k^p. For notational simplicity, the sets Υ_k from Section 4.3.1 and Υ_k^p are assumed distinct to differentiate between old transmitters and transmitters from the prior map.

The marginalized likelihoods for the transmitters in the prior map are calculated similarly to Eq. (4.47) as

$$p_{a_k^p} = p_{r,a_k^p} p_{v,a_k^p} \mathrm{p}_{e,\mathrm{a}}^{<a_k^p>}, \tag{5.39}$$

where

$$p_{r,a_k^p} = \sum_{\ell=1}^{N_{p,\mathrm{Tx}}^*} w^{<\cdot,a_k^p,\ell>} \, \mathrm{p}\left(\tilde{\boldsymbol{z}}_{\mathrm{R},k}^{\mathrm{new}} | \boldsymbol{x}_{\mathrm{TX}}^{<\cdot,a_k^p,\ell>}, a_k^p, \mathcal{A}_{k-1}, \boldsymbol{x}_{\mathrm{u},k}^{<i>}\right) \tag{5.40}$$

as in Eq. (4.46), and

$$p_{v,a_k^p} = \mathbb{E}_{h,k}^{<i,a_k^p>} \tag{5.41}$$

as in Eq. (4.42). The ℓ^{th} transmitter particle of the $a_k^{p\,\text{th}}$ transmitter in the prior map is denoted by $\boldsymbol{x}_{\mathrm{TX}}^{<\cdot,a_k^p,\ell>}$, and its weight by $w^{<\cdot,a_k^p,\ell>}$. The probability of existence $\mathrm{p}_{e,\mathrm{a}}^{<a_k^p>}$ represents the belief as in Eq. (5.7) that a transmitter in the prior map exists.

The set Γ_k for the case that both old transmitters estimated by the user so far and transmitters from the prior map that can be associated with the new transmitter is, similar to Eq. (4.48),

$$\Gamma_k = \{j : (j \in \Upsilon_k \vee j \in \Upsilon_k^p) \wedge p_j > \rho\}. \tag{5.42}$$

If multiple new transmitters are initialized, the greedy algorithm from Section 4.3.2 can be applied in a straightforward manner.

5.3 Map Merging

In Section 5.2, the use of a prior map has been explained with respect to one single user. This user first estimates the rotation and translation relating the coordinate systems of the user and the prior map and can then associate newly detected signal components with transmitters from the prior map. After the user has finished their run, the observations made by the user, i.e., the user's transmitter map, are merged with the prior map. This merging of the user map and the prior map is denoted by the term map merging. While merging the two maps, the user updates the prior map by adding information on transmitters obtained during their run. The new prior map after merging the user map into the prior map is denoted by the term *resulting map*. The resulting map can be shared with other users. As described in Section 5.1, information on a transmitter contains its state and visibility estimates, the probability of existence and a reliability distance.

Three cases are regarded for adding a transmitter from the user map into the resulting map as follows.

1. The transmitter has a correspondence in the prior map obtained from the map match as in Section 5.2.4.
2. The transmitter was associated with a transmitter from the prior map by at least one user-map particle.
3. The transmitter is not in the consensus set nor was it associated with a transmitter from the prior map by any user-map particle.

Case 1
If the ${j_\mathrm{u}}^{\mathrm{th}}$ transmitter in the user map corresponds to the ${j_\mathrm{p}}^{\mathrm{th}}$ transmitter in the prior map, there is a tuple $(j_\mathrm{u}, j_\mathrm{p})$ in the consensus set $\hat{\mathcal{C}}$. In this case, the information about the two transmitters is merged based on the reliability distances. The index of the transmitter in the resulting map after merging the ${j_\mathrm{u}}^{\mathrm{th}}$ transmitter in the user map and the ${j_\mathrm{p}}^{\mathrm{th}}$ transmitter in the prior map is assumed to be j. For merging the transmitters, the relative user reliability distance and the relative prior reliability distance are defined as

$$\bar{\mathrm{d}}_{\mathrm{rel,u}}^{<j>} = \frac{\mathrm{d}_{\mathrm{rel,u}}^{<j_\mathrm{u}>}}{\mathrm{d}_{\mathrm{rel,u}}^{<j_\mathrm{u}>} + \mathrm{d}_{\mathrm{rel,a}}^{<j_\mathrm{p}>}} \quad \text{and} \quad \bar{\mathrm{d}}_{\mathrm{rel,a}}^{<j>} = \frac{\mathrm{d}_{\mathrm{rel,a}}^{<j_\mathrm{p}>}}{\mathrm{d}_{\mathrm{rel,u}}^{<j_\mathrm{u}>} + \mathrm{d}_{\mathrm{rel,a}}^{<j_\mathrm{p}>}}, \tag{5.43}$$

respectively. With the relative reliability distances, the transmitter estimates from the user and the prior map can be merged with taking into account how much the user and the users who had created the prior map contributed in terms of observations made.

The state estimate PDF for the merged transmitter with index j in the resulting map, $\boldsymbol{x}_{\mathrm{TX}}^{<j>}$, is a particle cloud defined by

$$\begin{aligned} \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX}}^{<j>}\right) = \bar{\mathrm{d}}_{\mathrm{rel,u}}^{<j>} \sum_{\ell=1}^{N_{p,\mathrm{u}}^{<j_\mathrm{u}>}} w_k^{<i,j_\mathrm{u},\ell>} \delta\left(\boldsymbol{x}_{\mathrm{TX}}^{<j>} - \boldsymbol{x}_{\mathrm{TX},k}^{<i,j_\mathrm{u},\ell>}\right) \\ + \bar{\mathrm{d}}_{\mathrm{rel,a}}^{<j>} \sum_{\ell=1}^{N_{p,\mathrm{a}}^{<j_\mathrm{p}>}} w^{<j_\mathrm{p},\ell>} \delta\left(\boldsymbol{x}_{\mathrm{TX}}^{<j>} - \boldsymbol{x}_{\mathrm{TX}}^{<j_\mathrm{p},\ell>}\right), \end{aligned} \tag{5.44}$$

where $\boldsymbol{x}_{\mathrm{TX},k}^{<i,j_\mathrm{u},\ell>}$ is the ℓ^{th} of the $N_{p,\mathrm{u}}^{<j_\mathrm{u}>}$ transmitter particles of the ${j_\mathrm{u}}^{\mathrm{th}}$ transmitter of the i^{th} user-map particle in the user map, $w_k^{<i,j_\mathrm{u},\ell>}$ is its associated weight, $\boldsymbol{x}_{\mathrm{TX}}^{<j_\mathrm{p},\ell>}$ is the ℓ^{th} of the $N_{p,\mathrm{a}}^{<j_\mathrm{p}>}$ transmitter particles of the ${j_\mathrm{u}}^{\mathrm{th}}$ transmitter in the prior map and $w^{<j_\mathrm{p},\ell>}$ its associated weight.

The reliability distance of the merged transmitter is the sum of the reliability distances of the transmitters in the prior map and the user map, namely

$$\mathrm{d}_{\mathrm{rel}}^{<j>} = \mathrm{d}_{\mathrm{rel,a}}^{<j_\mathrm{p}>} + \mathrm{d}_{\mathrm{rel,u}}^{<j_\mathrm{u}>}. \tag{5.45}$$

The probability of existence $\mathrm{p}_e^{<j>}$ for the resulting j^{th} transmitter is calculated as

$$\mathrm{p}_e^{<j>} = \mathrm{p}_{e,\mathrm{u}}^{<j_\mathrm{u}>} \bar{\mathrm{d}}_{\mathrm{rel,u}}^{<j>} + \mathrm{p}_{e,\mathrm{a}}^{<j_\mathrm{p}>} \bar{\mathrm{d}}_{\mathrm{rel,a}}^{<j>}, \tag{5.46}$$

where $\mathrm{p}_{e,\mathrm{u}}^{<j_\mathrm{u}>}$ is the probability of existence of the transmitter in the user map, and $\mathrm{p}_{e,\mathrm{a}}^{<j_\mathrm{p}>}$ the probability of existence in the prior map.

The parameters of the Beta distributions in the visibility map in the h^{th} hexagon for the resulting j^{th} transmitter are

$$\begin{aligned} p_h^{<j>} &= p_{h,k}^{<j_\mathrm{u}>} + p_h^{<j_\mathrm{p}>}, \quad \text{and} \\ q_h^{<j>} &= q_{h,k}^{<j_\mathrm{u}>} + q_h^{<j_\mathrm{p}>}, \end{aligned} \tag{5.47}$$

where $p_{h,k}^{<j_\mathrm{u}>}$ and $q_{h,k}^{<j_\mathrm{u}>}$ are the corresponding parameters in the user map, and $p_h^{<j_\mathrm{p}>}$ and $q_h^{<j_\mathrm{p}>}$ in the prior map. The parameters $p_h^{<j>}$ and $q_h^{<j>}$ are independent from the reliability distances, which are inherently accounted for in the corresponding parameters of the user and the prior map.

Case 2

If a new signal component corresponding to the ${j_\mathrm{u}}^{\mathrm{th}}$ transmitter in the user map is associated with the ${j_\mathrm{p}}^{\mathrm{th}}$ transmitter from the prior map by a number of user-map particles, the information about the transmitter in the prior map is implicitly taken into the state estimate, the visibility map and the reliability distance for each single particle making the association. The transmitter is taken

into the resulting map if the thresholds for the transmitter's state uncertainty in Eq. (5.12) and the probability of existence τ_{ex} are met.

Case 3
A transmitter from the user map that is not associated is taken into the resulting map with its estimated state PDF, its visibility map, probability of existence and reliability distance as described in Section 5.1, if the thresholds for the transmitter's state uncertainty in Eq. (5.12) and the probability of existence τ_{ex} are met.

The transmitters from the prior map are in general taken into the resulting map. There are again three cases regarded for such a transmitter in the prior map as follows.

1. The transmitter has a correspondence in the user map obtained from the map match.

2. The transmitter was associated with a transmitter from the user map by at least one user-map particle.

3. The transmitter is not in the consensus set nor was it associated with a transmitter from the user map by any user-map particle.

Case 1
Such a case is handled by the first case described above. The information about the prior map transmitter is merged with the information about the transmitter from the user map and added to the resulting map.

Case 2
If there is an association between any new signal component, i.e., transmitter, and the $j_{\text{p}}{}^{\text{th}}$ transmitter from the prior map for a number of user-map particles, there are association tuples $(\cdot, j_{\text{p}})$ in the sets $\mathcal{A}_k^{<i>}$ of association decisions of the user-map particles. The transmitter j_{p} from the prior map is taken into the resulting map as the $\tilde{j}^{\text{th}}$ transmitter with its original state PDF, visibility map and reliability distance. Only the probability of existence is adapted. Denote the set of all user-map particles that have not associated the $j_{\text{p}}{}^{\text{th}}$ transmitter from the prior map with any other transmitter in the user map by $\tilde{\mathcal{T}}_{j_{\text{p}}}$. The probability of existence of the $\tilde{j}^{\text{th}}$ transmitter in the resulting map is set to

$$\text{p}_e^{<\tilde{j}>} = \sum_{i \in \tilde{\mathcal{T}}_{j_{\text{p}}}} w_k^{<i>}. \tag{5.48}$$

Thus, it is the summed weight of all particles that do not associate the original transmitter with a new transmitter. If all user-map particles decide for an association between the $j_{\text{p}}{}^{\text{th}}$ transmitter in the prior map and any new transmitter, the probability of existence of the corresponding transmitter in the resulting map is zero.

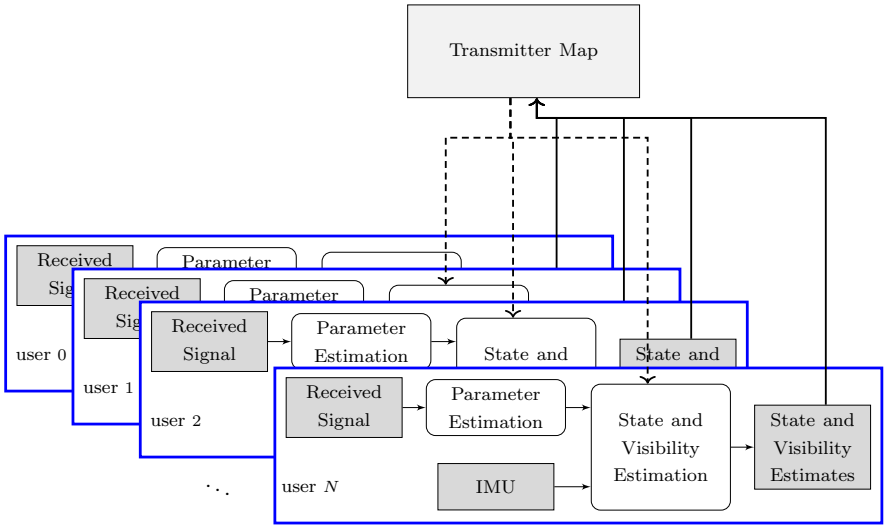


Figure 5.6: In Channel-SLAM with cooperation among users, estimates on the transmitter states and visibilities are used to update the transmitter map that is handed from one user to another.

Case 3
In the case of no association during the map match or with any user transmitter, a transmitter from the prior map is taken into the resulting map as is.

After creating the resulting map, transmitters that do not meet the thresholds regarding their state uncertainty in Eq. (5.12) and probability of existence τ_{ex} are deleted from the resulting map.

Finally, the resulting map is distributed to other users. A flow chart of the Channel-SLAM algorithm with cooperating users is depicted in Fig. 5.6. The estimates on the transmitter states and visibilities as output of single user Channel-SLAM depicted in Fig. 3.2 are used to update the resulting transmitter map that is handed from one user to another to increase the localization performance. The transmitter map may be shared directly among users or via a central entity.

5.4 Summary and Outlook

Cooperation among users is a paradigm shift in Channel-SLAM. We have seen how users can create, share and update transmitter maps. Such cooperation increases the positioning performance of Channel-SLAM considerably. Transmitter visibility information plays a crucial role, as it permits robust map matching, which is a requirement for sharing transmitter maps.

Since the estimated transmitter state PDFs are represented by particle clouds in Channel-SLAM, the memory consumption of transmitter maps can be high.

Hence, these estimated state PDFs need to be transformed into PDFs with lower memory consumption when transmitter maps are exchanged. In the following chapter, we will derive a new tracking filter that can represent transmitter states as Gaussian mixture models. Such models can typically be expressed with only few parameters.

Chapter 6

Channel-SLAM with Gaussian Sum Filters

The Channel-SLAM algorithm derived in Chapter 4 uses a Rao-Blackwellized particle filter to estimate the user state jointly with the transmitter states and visibilities. The state PDF of a transmitter is thus represented by a possibly large number of particles. When maps of transmitter states are shared among users, all particles and their weights need to be transmitted. It is therefore preferable to represent transmitter state PDFs in a parametrized fashion, while keeping the transmitted information the same. The Gaussian sum particle filter (GSPF) presented within this section is a novel filter for Channel-SLAM, in which the transmitter states are estimated by Gaussian sum filters. This representation reduces the memory complexity considerably.

The idea and structure of the GSPF are presented in Section 6.1. The filter is derived in Section 6.2. Section 6.3 presents a data association scheme for the GSPF.

6.1 The Idea and Structure of the Gaussian Sum Particle Filter

The idea behind the GSPF is to use a parametrized PDF instead of a particle cloud to represent a transmitter state estimate. A parametrized PDF can be stored more efficiently, and consequently allows to decrease the amount of data that needs to be transmitted when maps of transmitter states are shared. While a Gaussian distribution can be represented very efficiently with a mean and a covariance matrix, it does very often not match the targeted posterior PDFs of the transmitter states, which may be of any shape and composed of different modes. For example, the shape of a PDF of a transmitter location estimate after initialization with a ToA and an AoA measurement in Channel-SLAM resembles a circular sector as in Fig. 3.11. With Gaussian mixture models, PDFs composed of different modes can be well approximated with relatively

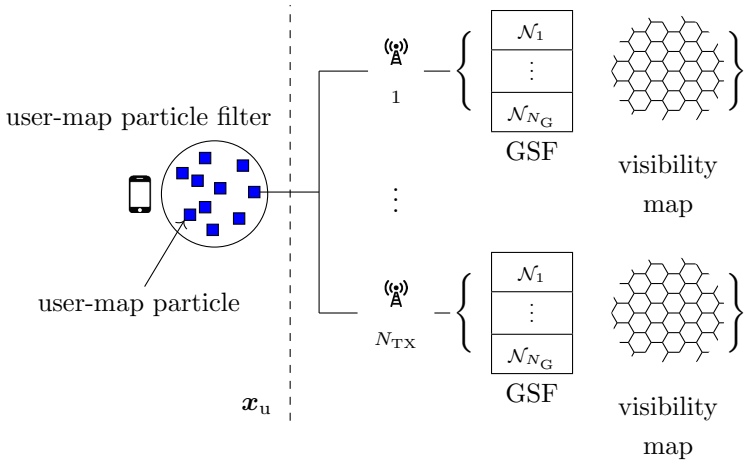


Figure 6.1: The structure of the Gaussian sum particle filter for Channel-SLAM. The posterior PDFs of the transmitter states are represented by Gaussian mixture models for every user-map particle, and estimated by Gaussian sum filters. The components of the Gaussian mixtures are estimated by unscented Kalman filters. A hexagonal visibility map is estimated with every user-map particle.

few parameters.

One approach is to perform Channel-SLAM with the Rao-Blackwellized particle filter as described in Section 4.2, and to approximate the estimated posterior PDF of every transmitter state, which is represented by a particle cloud as in Eq. (4.14), by a Gaussian mixture model when a map is exchanged. Estimating a parametrized PDF from a set of samples, or particles, is a problem referred to as density estimation [Sil86].

The approach of the GSPF for Channel-SLAM is to describe the transmitter state PDFs with Gaussian mixture models. While the user state is still estimated with a particle filter, the single Gaussian components in the Gaussian mixture are estimated with unscented Kalman filters. The structure of the GSPF is illustrated in Fig. 6.1. Following Eqs. (4.10) and (4.12), the state of each transmitter is estimated for every user-map particle, and the transmitter states are estimated independently from each other assuming independent measurements. The state PDF of the j^{th} transmitter of the i^{th} user-map particle at time instants zero to k corresponding to Eq. (4.14) is modeled by a Gaussian mixture of N_G weighted components. It is expressed as

$$\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{R},1:k}\right) = \sum_{\ell=1}^{N_\mathrm{G}} w_{0:k}^{<i,j,\ell>} \mathcal{N}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j>}; \boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j,\ell>}, \boldsymbol{P}_{0:k}^{<i,j,\ell>}\right), \quad (6.1)$$

where the history of the ℓ^{th} Gaussian component has a mean $\boldsymbol{x}_{\mathrm{TX},0:k}^{<i,j,\ell>}$, a covariance matrix $\boldsymbol{P}_{0:k}^{<i,j,\ell>}$ and an associated weight $w_{0:k}^{<i,j,\ell>}$. The mean and covariance matrix at time instant k are obtained from the update of the cor-

responding unscented Kalman filter at the previous time instant $k-1$. The unscented Kalman filter equations are presented in Appendix A.2. Although N_G depends on the time instant k, the user-map particle index i and the transmitter index j, these indices are omitted in N_G for notational simplicity.

6.2 The Gaussian Sum Particle Filter for Channel-SLAM

The prediction step for the user-map particle filter in the GSPF case is the same as for the user-map particle filter in Section 4.2, as the importance density is the same. A new user-map particle is thus drawn from the importance density in Eq. (4.25). The likelihood referring to the radio measurement for the j^{th} transmitter in the update step is calculated by

$$\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{z}_{\mathrm{V},k}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}\right) = \sum_{\ell=1}^{N_\mathrm{G}} w_k^{<i,j,\ell>} \mathcal{N}\left(\boldsymbol{z}_{\mathrm{R},k}^{<j>}; \hat{\boldsymbol{z}}_{\mathrm{R},k}^{<i,j,\ell>}, \boldsymbol{R}_k^{<j>}\right). \tag{6.2}$$

The measurement covariance matrix of the j^{th} transmitter at time instant k is denoted by $\boldsymbol{R}_k^{<j>}$. In Channel-SLAM, it is a diagonal matrix with the variances ${\sigma_{\tau,k}^{<j>}}^2$ and ${\sigma_{\theta,k}^{<j>}}^2$ of the ToA and AoA measurements on the diagonal. The radio likelihood of the ℓ^{th} component in Eq. (6.2) is then

$$\begin{aligned}\mathcal{N}\left(\boldsymbol{z}_{\mathrm{R},k}^{<j>}; \hat{\boldsymbol{z}}_{\mathrm{R},k}^{<i,j,\ell>}, \boldsymbol{R}_k^{<j>}\right) = \mathcal{N}\left(\tau_k^{<j>}; \hat{\tau}_k^{<i,j,\ell>}, {\sigma_{\tau,k}^{<j>}}^2\right) \\ \times \mathcal{N}\left(\theta_k^{<j>}; \hat{\theta}_k^{<i,j,\ell>}, {\sigma_{\theta,k}^{<j>}}^2\right).\end{aligned} \tag{6.3}$$

The predicted measurement $\hat{\boldsymbol{z}}_{\mathrm{R},k}^{<i,j,\ell>}$ is a vector of the predicted ToA and AoA, $\hat{\tau}_k^{<i,j,\ell>}$ and $\hat{\theta}_k^{<i,j,\ell>}$, that can be calculated analogue to Eq. (3.23) and Eq. (3.24), respectively, with the mean of the ℓ^{th} Gaussian component.

The prediction and update equations of the unscented Kalman filter are calculated by the standard unscented Kalman filter equations presented in Appendix A.2. The crucial part in the GSPF is the recursive calculation of the weights in the user-map particle filter. Following Eq. (4.32), the weights in the user-map particle filter at time instant k are

$$w_k^{<i>} \propto w_{k-1}^{<i>} \, \mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{M}_{k-1}^{<i>}\right) \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right). \tag{6.4}$$

The second factor on the right hand side of Eq. (6.4) can be calculated as in Eqs. (4.35) and (4.36). The third factor can be written with the assumption of

independent transmitter states and a first order Markov model as

$$\begin{aligned}
&\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\
&\quad = \prod_{j=1}^{N_{\mathrm{TX},k}} \int \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}\right) \qquad (6.5) \\
&\qquad\qquad \times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \mathrm{d}\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>} \\
&\quad = \prod_{j=1}^{N_{\mathrm{TX},k}} \int \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{z}_{\mathrm{V},k}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}\right) \\
&\qquad\qquad \times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \mathrm{d}\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}. \qquad (6.6)
\end{aligned}$$

The first factor in the integral in Eq. (6.6) is the likelihood in Eq. (6.2). For notational simplicity, the likelihood of the j^{th} transmitter with respect to the ℓ^{th} Gaussian component for the i^{th} user-map particle as in Eq. (6.3) is defined as

$$\boldsymbol{g}_z\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right) \coloneqq \mathcal{N}\Big(\tau_k^{<j>}; \hat{\tau}_k^{<i,j,\ell>}, {\sigma_{\tau,k}^{<j>}}^2\Big)\mathcal{N}\Big(\theta_k^{<j>}; \hat{\theta}_k^{<i,j,\ell>}, {\sigma_{\theta,k}^{<j>}}^2\Big). \qquad (6.7)$$

The second factor in the integral in Eq. (6.6) describes the state of the j^{th} transmitter for the i^{th} user-map particle after the prediction and before the update step at time instant k. The corresponding state of the ℓ^{th} Gaussian component is defined as

$$\mathsf{p}_{\mathrm{TX},k}^{<i,j,\ell>-} \coloneqq \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) = \mathcal{N}\Big(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}; \boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>-}, \boldsymbol{P}_k^{<i,j,\ell>-}\Big). \qquad (6.8)$$

The integrand in Eq. (6.6) can then be written with Eqs. (6.7) and (6.8) as

$$\begin{aligned}
&\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{z}_{\mathrm{V},k}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}\right) \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\
&\quad = \sum_{\ell=1}^{N_{\mathrm{G}}} w_{k-1}^{<i,j,\ell>} \mathsf{p}_{\mathrm{TX},k}^{<i,j,\ell>-} \boldsymbol{g}_z\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right). \qquad (6.9)
\end{aligned}$$

Substituting Eq. (6.9) in Eq. (6.6) leads to

$$\begin{aligned}
&\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\
&\quad = \prod_{j=1}^{N_{\mathrm{TX},k}} \sum_{\ell=1}^{N_{\mathrm{G}}} w_{k-1}^{<i,j,\ell>} \int \mathsf{p}_{\mathrm{TX},k}^{<i,j,\ell>-} \boldsymbol{g}_z\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right) \mathrm{d}\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}. \qquad (6.10)
\end{aligned}$$

The second factor in the above integral as in Eq. (6.7) is a nonlinear function in the integration variable $\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}$. However, the integral has the form of

Eq. (2.18). Thus, it can be approximated with the unscented transform as

$$\begin{aligned}
&\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\
&= \prod_{j=1}^{N_{\mathrm{TX},k}} \sum_{\ell=1}^{N_{\mathrm{G}}} w_{k-1}^{<i,j,\ell>} \int \mathsf{p}_{\mathrm{TX},k}^{<i,j,\ell>-} \boldsymbol{g}_z\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right) \mathrm{d}\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>} \\
&= \prod_{j=1}^{N_{\mathrm{TX},k}} \sum_{\ell=1}^{N_{\mathrm{G}}} w_{k-1}^{<i,j,\ell>} \sum_{m=1}^{N_{\mathrm{sig}}} \omega_m \, \boldsymbol{g}_z\left(\boldsymbol{X}_m\right),
\end{aligned} \tag{6.11}$$

where $\boldsymbol{X}_m$ is the m^{th} of the N_{sig} sigma points and ω_m its associated weight. The sigma points and weights can be calculated by Eq. (2.19), for example, where

$$\boldsymbol{\mu}_{\boldsymbol{x}} = \boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>-} \quad \text{and} \quad \boldsymbol{C}_{\boldsymbol{x}} = \boldsymbol{P}_k^{<i,j,\ell>-} \tag{6.12}$$

are the mean and the covariance matrix, respectively, obtained from the corresponding unscented Kalman filter before the update, and $N = 3$ is the dimension of a transmitter's state. The indices of the transmitters and Gaussian components as well as of the time instants are omitted in the sigma points and their weights for notational simplicity. The weights in Eq. (6.4) are calculated with Eqs. (4.35) and Eq. (4.36) as

$$w_k^{<i>} \propto w_{k-1}^{<i>} \prod_{j=1}^{N_{\mathrm{TX},k}} \mathrm{p}\left(\boldsymbol{z}_{\mathrm{V},k}|\boldsymbol{M}_{h,k-1}^{<i,j>}\right) \sum_{\ell=1}^{N_{\mathrm{G}}} w_{k-1}^{<i,j,\ell>} \sum_{m=1}^{N_{\mathrm{sig}}} \omega_m \, \boldsymbol{g}_z\left(\boldsymbol{X}_m\right). \tag{6.13}$$

From Eq. (6.11) follows for the weight update of the unscented Kalman filter for the ℓ^{th} Gaussian component of the j^{th} transmitter

$$w_k^{<i,j,\ell>} \propto w_{k-1}^{<i,j,\ell>} \sum_{m=1}^{N_{\mathrm{sig}}} \omega_m \, \boldsymbol{g}_z\left(\boldsymbol{X}_m\right). \tag{6.14}$$

After the update of the weights in Eq. (6.11) and Eq. (6.14), they are normalized to one by

$$w_k^{<i,j,\ell>} = \frac{w_k^{<i,j,\ell>}}{\sum\limits_{\tilde{\ell}=1}^{N_{\mathrm{G}}} w_k^{<i,j,\tilde{\ell}>}} \quad \text{and} \quad w_k^{<i>} = \frac{w_k^{<i>}}{\sum\limits_{\tilde{i}=1}^{N_p} w_k^{<\tilde{i}>}}. \tag{6.15}$$

The mean of a Gaussian component can be regarded as a hypothesis of the state of the transmitter, and its covariance as the corresponding uncertainty. The associated weight indicates how likely this hypothesis is. As the user moves, the means, covariances and weights of the Gaussian components evolve. When the weight of a Gaussian component falls below a threshold, the corresponding Gaussian component is deleted from the Gaussian mixture to decrease the computational complexity.

6.3 Data Association

In the following, we derive a data association scheme for the GSPF analogue to Section 4.3.

Eq. (4.40) holds for the GSPF case as well. The second factor in Eq. (4.40) can be calculated in the same way as for the Rao-Blackwellized particle filter with Eq. (4.41) and Eq. (4.42). For the first factor in Eq. (4.40), Eq. (4.43) and Eq. (4.44) hold.

In the GSPF case, the second factor in the integral of Eq. (4.43) is the PDF of the $a_k{}^{\text{th}}$ transmitter after the prediction and before the update in the Gaussian sum filter. It is therefore

$$\begin{aligned} &\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k>} | a_k, \mathcal{A}_{k-1}, \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\ &\quad = \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{k-1}^{<i,a_k,\ell>} \mathcal{N}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k>}; \boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k,\ell>-}, \boldsymbol{P}_k^{<i,a_k,\ell>-}\right), \end{aligned} \tag{6.16}$$

where $\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k,\ell>-}$ and $\boldsymbol{P}_k^{<i,a_k,\ell>-}$ are predicted the mean and covariance of the ℓ^{th} component of the Gaussian mixture of the $a_k{}^{\text{th}}$ transmitter, and $w_{k-1}^{<i,a_k,\ell>}$ the corresponding weight. For notational simplicity, the ℓ^{th} addend in Eq. (6.16) is denoted by $\mathsf{p}_{\mathrm{TX},k}^{<i,a_k,\ell>-}$ similar to Eq. (6.8),

$$\mathsf{p}_{\mathrm{TX},k}^{<i,a_k,\ell>-} := \mathcal{N}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k>}; \boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k,\ell>-}, \boldsymbol{P}_k^{<i,a_k,\ell>-}\right). \tag{6.17}$$

Analogue to Eq. (6.7), the corresponding likelihood is denoted by

$$\boldsymbol{g}_z\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k,\ell>}\right) := \mathcal{N}\left(\tau_{j,k}; \hat{\tau}_k^{<i,a_k,\ell>}, \sigma_{\tau,k}^{<j>2}\right) \mathcal{N}\left(\theta_{a_k,k}; \hat{\theta}_k^{<i,a_k,\ell>}, \sigma_{\theta,k}^{<j>2}\right). \tag{6.18}$$

Substituting Eq. (6.17) into Eq. (6.16) and the result and Eq. (6.18) into Eq. (4.43) yields

$$\begin{aligned} p_{r,a_k} &= \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{k-1}^{<i,a_k,\ell>} \int \mathsf{p}_{\mathrm{TX},k}^{<i,a_k,\ell>-} \boldsymbol{g}_z\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k,\ell>}\right) \mathrm{d}\boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k,\ell>} \\ &= \sum_{\ell=1}^{N_{p,\mathrm{Tx}}} w_{k-1}^{<i,a_k,\ell>} \sum_{m=1}^{N_{\mathrm{sig}}} \omega_m \, \boldsymbol{g}_z\left(\boldsymbol{X}_m\right) \end{aligned} \tag{6.19}$$

with the unscented transform, where the sigma points $\boldsymbol{X}_m$ and their weights ω_m can again be obtained by Eq. (2.19), for example, where

$$\boldsymbol{\mu}_{\boldsymbol{x}} = \boldsymbol{x}_{\mathrm{TX},k}^{<i,a_k,\ell>-} \quad \text{and} \quad \boldsymbol{C}_{\boldsymbol{x}} = \boldsymbol{P}_k^{<i,a_k,\ell>-} \tag{6.20}$$

are the predicted mean and the covariance, respectively, from the corresponding unscented Kalman filter. Finally, the marginalized likelihoods are calculated as

$$p_{a_k} = p_{r,a_k} p_{v,a_k} \tag{6.21}$$

with p_{r,a_k} from Eq. (6.19) and p_{v,a_k} from Eq. (4.42).

If only one new transmitter is detected at time instant k, the ML method or data association sampling can be used with the marginalized likelihoods from Eq. (6.21). In the case of multiple transmitters, the greedy algorithm from Section 4.3.2 is applied. If transmitters from a prior map may be incorporated, the method from Section 5.2.9 is applied in a straightforward manner.

6.4 Summary and Outlook

The GSPF allows for a compact representation of transmitter states in the transmitter tracking filter. The communication overhead when sharing transmitter maps is considerably reduced without additional signal processing in terms of changing the representation of transmitter state estimates.

The following chapter will evaluate the methods derived in this thesis by simulations.

Chapter 7

Performance Evaluations

While the concept of Channel-SLAM and virtual transmitters has been shown to work in real indoor [Gen+16a; Gen+17a; GU17; KG20] and outdoor [Gen+14; Gen+15; Gen+17b; Ulm+18] scenarios, simulations in an indoor environment are performed to evaluate the methods developed in this thesis.

The simulation scenario and relevant parameters of the simulations are introduced in Section 7.1. Section 7.2 briefly discusses the implementation of the simulations. The results are presented in Section 7.3.

7.1 Simulation Scenario

The simulation scenario is depicted in Fig. 7.1. It shows the top view of an indoor shopping mall with one single physical transmitter marked by the red triangle labeled Tx. The thick black lines represent walls that reflect the transmit signal, and the black dots model point scatterers.

The physical transmitter continuously transmits a signal with a uniform power spectrum density using an omnidirectional antenna. The bandwidth of the signal is 100 MHz and the carrier frequency 1.9 GHz. While a high bandwidth is preferred for multipath assisted positioning schemes as they rely on the ability to resolve MPCs at the receiver, the bandwidth of 100 MHz was chosen since it is in accordance with current mass market technology. For example, the 3rd Generation Partnership Project (3GPP) Long Term Evolution (LTE) standard provides bandwidths up to 100 MHz with carrier aggregation [3rd13]. Likewise, the IEEE 802.11 WLAN standard defines signal bandwidths of up to 80 MHz and optionally up to 160 MHz [IEE16].

With the locations of the physical transmitter and the walls and scatterers, a channel impulse response can be calculated at each location in the scenario. A power loss of 3 dB is assumed when the signal is reflected by a wall. When a signal is scattered at a point scatterer, the power loss is assumed 6 dB. The noise-free channel impulse response is band-limited to 100 MHz, and white Gaussian noise is added to obtain the received signal. In the following evaluations, propagation paths up to an order of three are considered. Thus, the transmit

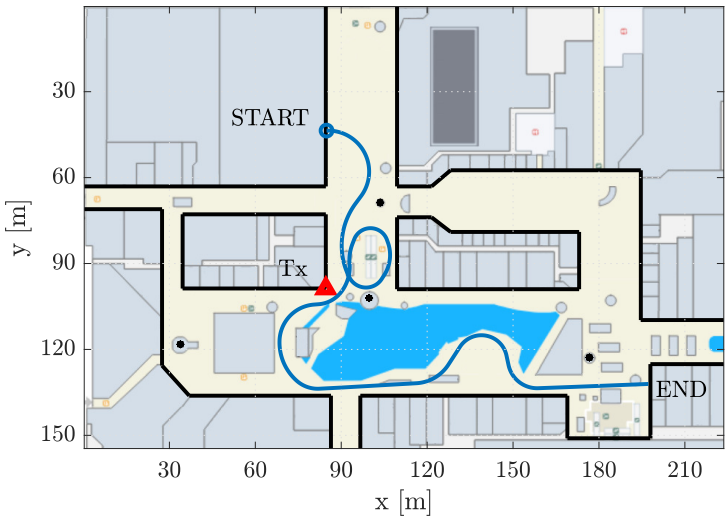


Figure 7.1: Top view of an indoor mall serving as the simulation scenario with one physical transmitter marked by the red triangle labeled Tx. Black lines represent walls that reflect the transmit signal, whereas black dots represent point scatterers. A user track is depicted by the blue line.

signal from the physical transmitter may be reflected and scattered at up to three objects before arriving at the user.

Each user is equipped with a planar antenna array consisting of nine elements aligned in a uniform quadratic grid. The distance between neighboring antenna elements is a quarter of a wavelength.

The transmit signal is known to the user. Every 100 ms, a snapshot of the received signal is sampled at the receiver and recorded. The sampled snapshot serves then as input in KEST for tracking the signal components.

Neither the location of the physical transmitter nor the locations of the walls and scatterers are known to the user. Consequently, the states of the virtual transmitters are unknown as well.

The user carries an inertial measurement unit, which is rigidly mounted on the receiver. However, no acceleration measurements, but only rates of change of the user heading from the gyroscope are used. They are incorporated in the user movement model as described in Section 3.3.

In Fig. 7.1, there is one user track depicted by the blue line. The start and end points of the track are labeled START and END, respectively. This track serves as a reference for some of the following evaluations and is denoted by the term *reference track*. The length of the reference track is 311.2 m.

Unless stated otherwise, the following parameters are used for the evaluations.

- Signal
  - Bandwidth: 100 MHz
  - Carrier frequency: 1.9 GHz
- Particle Filter
  - Number of user and user-map particles: $N_p = 500$, see Eqs. (3.16) and (4.13)
  - Effective sample size threshold: $\bar{N}_p^{\text{eff}} = 0.8 N_p$, see Section 2.2.3.4
  - Time between time instants: $T_k = 100\,\text{ms} \;\; \forall\, k > 1$, see Section 3.5.2
- Visibility mapping
  - Prior Beta distribution parameters: $\nu_h^{<j>} = \bar{\nu}_h^{<j>} = 0.8$, see Eq. (4.5)
  - Hexagon side length: 2 m, see Section 4.1
- Data association
  - Threshold for association: $\rho = 10^{-4}$, see Eqs. (4.48) and (4.56)
  - Likelihood for no association: $p_0 = 10^{-3}$, see Section 4.3.1
  - Likelihood for no association: $p_{(0,0)} = 10^{-3}$, see Section 4.3.2
  - Method: data association sampling, see Section 4.3
- Map creation

  - Threshold transmitter covariance: $\tau_\sigma = 20\,\mathrm{m}^2$, see Eq. (5.12)
  - Threshold transmitter probability of existence: $\tau_{\mathrm{ex}} = 0.2$, see Section 5.1

- Map matching
  - RANSAC success probability: $p_{\mathrm{succ}} = 0.95$, see Eq. (5.13)
  - RANSAC distance threshold: $\delta_d = 6\,\mathrm{m}$, see Eq. (5.31)
  - Reward term: $\rho_{\mathrm{rew}} = 2\,\mathrm{m}$, see Eq. (5.32)
  - Summed squared distance threshold: $\tau_{\mathrm{dist}} = 100\,\mathrm{m}^2$, see Section 5.2.8
  - Hypothesis error threshold: $\tau_E = 10\,\mathrm{m}$, see Section 5.2.8
  - Map matching interval: $d_{\mathrm{int}} = 2\,\mathrm{m}$, see Section 5.2.8

7.2 Implementation

In the channel estimator KEST, i.e., the first step of Channel-SLAM, the Kalman filters in the outer stage are implemented in MATLAB©. The SAGE algorithm in the inner stage is implemented in C++.

The environment of the state and visibility estimation, i.e., the second step in Channel-SLAM, is written in MATLAB©. The implementation of the actual filtering process at each time instant including the Rao-Blackwellized particle filter, data association and the map matching algorithm, is written in C++. As an interface between MATLAB© and the C++ functions, MEX files are used.

Since this thesis focuses on the second step in Channel-SLAM, all evaluations in terms of complexity refer to the second step of Channel-SLAM and its implementation in C++. Overhead of passing variables between MATLAB© and C++ is neglected.

All simulations were performed on a single core of an Intel© Xeon© processor E5-2697 with a processor frequency of 2.3 GHz.

7.3 Simulation Results

In the following, the methods developed in the thesis are evaluated regarding the positioning performance and the complexity. Unless stated otherwise, it is assumed that the starting location and heading of each user are known in a local coordinate system. However, no relation of the local coordinate system to any other reference system is known. Furthermore, no knowledge on the states or visibilities of the physical and virtual transmitters is assumed. The simulation scenario itself, i.e., the location of the walls and scatterers, is unknown as well. The users move with a speed of $1\,\mathrm{m/s}$.

The positioning performance is evaluated in terms of the mean absolute error (MAE) of the user position. The MAE is defined at time instant k as

$$\mathrm{MAE}_k = \sum_{i=1}^{N_p} w_k^{<i>} \|\underline{\boldsymbol{p}}_{\mathrm{u},k} - \boldsymbol{p}_{\mathrm{u},k}^{<i>}\|, \tag{7.1}$$

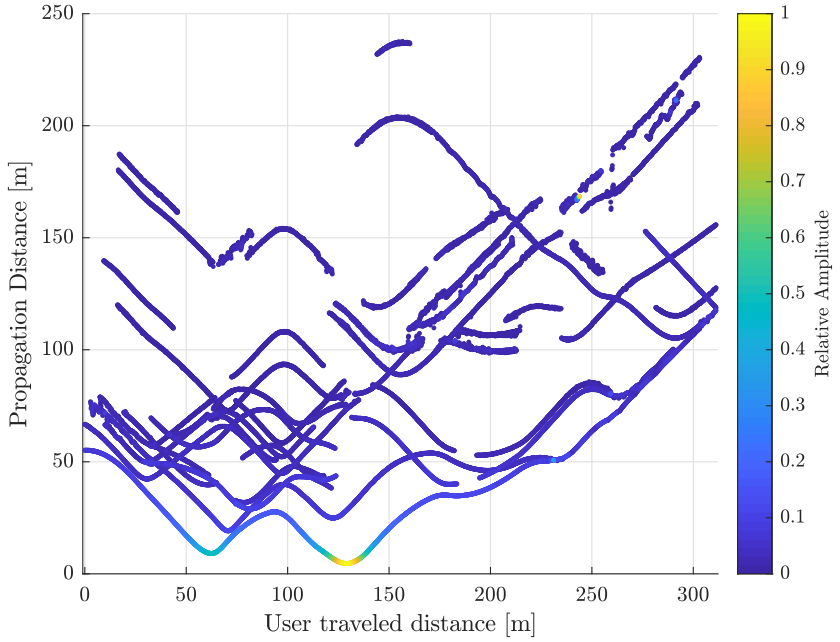


Figure 7.2: The results of KEST for the reference track. The propagation distances of signal components are plotted versus the distance traveled by the user. The color of the curves indicates the relative amplitude of the signal components.

where $\boldsymbol{p}_{\mathrm{u},k}^{<i>}$ is the position and $w_k^{<i>}$ the weight of the i^{th} user particle, and $\underline{\boldsymbol{p}}_{\mathrm{u},k}$ the true position of the user. Since particle filters are Monte Carlo methods and therefore probabilistic, all plots showing MAE curves are averaged over 250 iterations.

7.3.1 Reference Track

The KEST results for the reference track are plotted in Fig. 7.2. The plot shows the propagation distances of signal components versus the traveled distance of the user. For each time instant corresponding to a traveled distance on the horizontal axis, the vertical axis yields propagation distances of signal components estimated by KEST. The colors of the curves indicate the absolute values of the amplitudes of the signal components relative to the maximum amplitude of all signal components throughout the track in linear domain.

The curve with the shortest propagation distance throughout the track corresponds to the LoS path. As can be seen in Fig. 7.1, the user is constantly in LoS to the physical transmitter, and the LoS component is tracked throughout the entire track. The propagation distance of the LoS component is 55.1 m at the beginning of the track, and 117.31 m at the end, corresponding to the distance of the user from the physical transmitter at the respective positions.

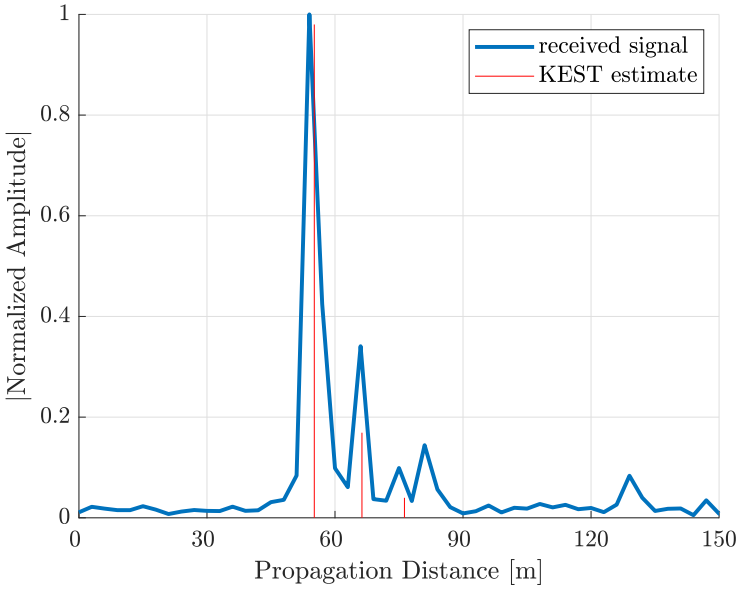


Figure 7.3: The results of KEST for the reference track at one snapshot at a traveled distance of 4 m. The blue curve shows the absolute value of the amplitude of the received signal versus the propagation distance. The red lines indicate the KEST estimates regarding the propagation distances and amplitudes. The amplitudes are normalized to the maximum amplitude of the received signal.

The SNR at the receiver averaged over the track is 13.0 dB. Overall, $N_{\mathrm{TX}}^{\mathrm{a}} = 47$ signal components and thus transmitters are detected by KEST.

A snapshot of the KEST algorithm at a traveled distance of 4 m along the reference track in Fig. 7.1 is illustrated in Fig. 7.3. It shows the absolute value of the amplitude of the received signal drawn in blue versus the propagation distance. The KEST estimates for the amplitudes and the ToAs corresponding to propagation distances of signal components are drawn in red. The amplitudes are normalized to the maximum amplitude of the received signal. One can see that at the corresponding user position, three signal components, i.e., transmitters, are tracked by KEST.

The user MAE for the reference track is plotted in blue in Fig. 7.4 versus the traveled distance. As mentioned above, the MAE is averaged over 250 simulation runs. The data association scheme from Section 4.3 is applied, but no mapping of the transmitter visibilities. Thus, $p_{v,(a,\nu)}$ is set to 1 in Eq. (4.53).

In addition to the MAE, the 95^{th} and 5^{th} percentiles are plotted in Fig. 7.4. In 95% of the simulation runs, the error was below the dotted line, while in 5% of the runs, it was below the dashed line.

The starting location of the user is assumed to be known in a local coordinate system. The initial user MAE is therefore zero. The MAE grows in the

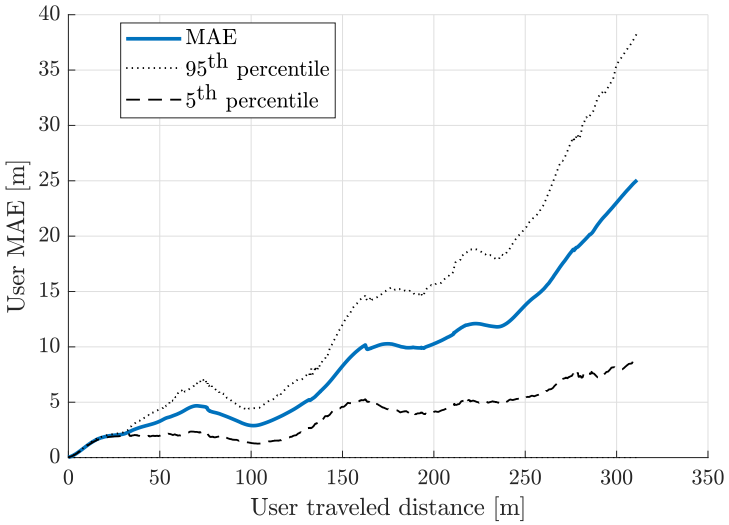


Figure 7.4: The MAE is plotted in blue for the reference track in Fig. 7.1 versus the distance the user has traveled. In addition, the 95^{th} and 5^{th} percentiles are plotted with a dotted and a dashed line, respectively.

beginning, since the states of the transmitters that are initialized based on newly detected signal components tend to have a high uncertainty in their estimates. However, when the user travels along the loop, data association can establish associations among newly detected signal components and old transmitters. Old transmitters are transmitters that had been observed earlier. Thus, the positioning error is limited and even shrinks after a traveled distance of approx. 72 m.

Leaving the loop, the positioning error increases again, since many new transmitters are detected that had not been visible before. Consequently, the data association scheme can not contribute. In particular during the last 70 m of the user track, the MAE grows rapidly. This can be explained with the high geometrical dilution of precision in this region. Almost all signal components arrive at the user from a similar direction.

The error curve in Fig. 7.4 shows a fundamental issue in Channel-SLAM. The positioning error is limited when a user can reuse transmitters with a data association scheme and detect loop closures as during the first approx. 120 m on the reference track. When the user enters a new area where no loop closures can be exploited, though, the contribution of data association schemes is small and the positioning error keeps increasing as new transmitters are detected and track of old ones is lost.

This issue motivates the need for cooperation among users. If many users cooperate by creating a comprehensive transmitter map, newly detected signal components can be associated with transmitters in the map and thus be ini-

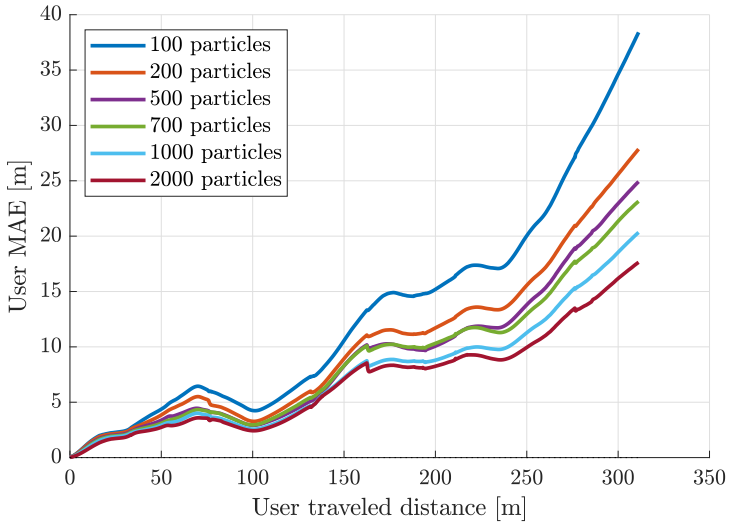


Figure 7.5: The MAE is plotted for the reference track in Fig. 7.1 for different numbers of user particles.

tialized with a low uncertainty about their state. The MAE curve is expected to flatten out in this case. Users may not revisit places where they themselves had been before, but visit places where other users have been and mapped transmitter states and visibilities already.

To understand the influence of the motion model and the gyroscope on the Channel-SLAM performance, a user walks along the reference track estimating their own position only based on the motion model and the gyroscope. Hence, in the user particle filter, only the prediction step is performed, but not the update step, and the user makes no use of any transmitters.

Even if the initial velocity is known, the MAE grows quadratically to a value in the order of 320 m at the end of the track. The motion model and the gyroscope do only have short term influence on the positioning performance, in particular at the beginning of a track, where the uncertainty about all detected transmitters' states is still high.

7.3.2 Number of Particles and Complexity

The number of user particles influences the positioning performance, but also the complexity of Channel-SLAM. Fig. 7.5 shows the MAE for the reference track in Fig. 7.1 for different numbers of particles.

Fig. 7.6 shows the processing time for the Rao-Blackwellized particle filter in the Channel-SLAM algorithm at each time instant for different numbers of user particles. The processing time from the channel estimator is neglected in these evaluations. Using KEST, this processing time is on average approx. 560 ms

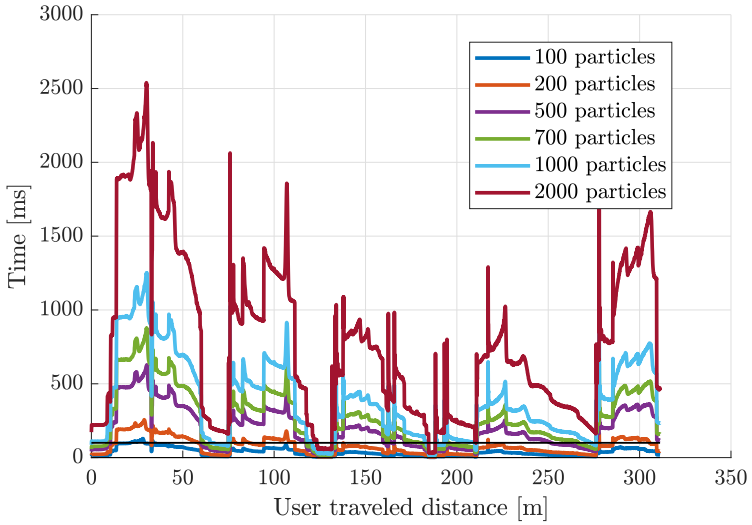


Figure 7.6: The processing times of Channel-SLAM are plotted for the user track in Fig. 7.1 for different numbers of user particles.

per time instant for the reference track. No visibility mapping or map matching were performed. The black line indicates the time between time instants in the Channel-SLAM implementation, which is 100 ms. Thus, if the processing time curves are below the black line, the filter runs in real-time. For the blue curve representing 100 user particles, the filter runs in real-time almost throughout the user track. The purple line representing 500 user particles indicates that the filter runs in real-time during only 90.6 m of the 311.2 m of traveled distance. The plot also shows how the computational complexity of Channel-SLAM grows linearly with the number of particles.

The processing time of Channel-SLAM at one time instant depends on the number of transmitters $N_{\mathrm{TX},k}$ and the numbers of transmitter particles $N_{p,\mathrm{Tx}}$ for the transmitters. The number of transmitter particles depends on the transmitter and the time instant. In addition, this number may be different for different user-map particles in the Rao-Blackwellized particle filter. It tends to be high upon transmitter initialization, in particular if the corresponding transmitter is located far from the user and if the uncertainty regarding the KEST parameter estimates is high. Thus, the particle reduction method from [Gen+17b] mentioned in Section 3.6 is used, where the number of transmitter particles is adapted to the uncertainty about a transmitter's state estimate.

The maximum number of transmitter particles observed in the simulations for one transmitter and one user particle was 864. Averaging over the user-map particles in the reference track, the mean of the number of transmitter particles for that transmitter while it is visible was 492. The physical transmitter is initialized with 89 particles for each user particle on the reference track. This

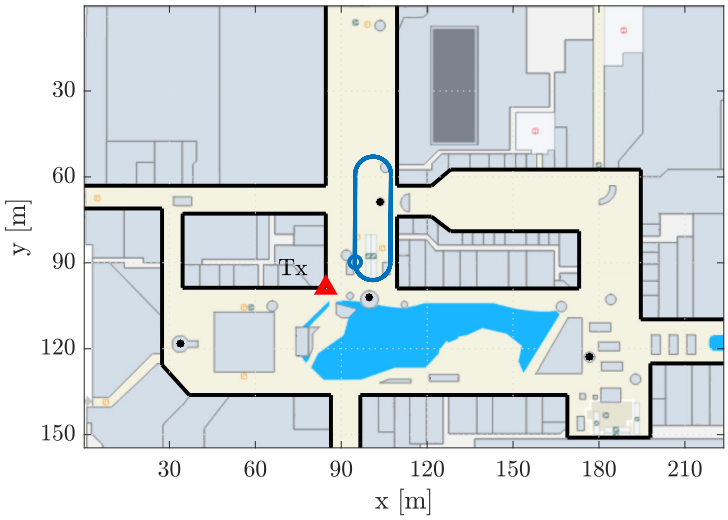


Figure 7.7: The simulation scenario with the track to evaluate the data association methods. The starting point is marked with the circle, and the user heads towards the top walking along the track for four laps.

number drops on average down to 11 after a traveled distance of approx. 77 m, and stays at 10 after a traveled distance of approx. 139 m.

7.3.3 Data Association

To evaluate the impact of data association, we regard the user track depicted in Fig. 7.7. The scenario itself is the same as in Fig. 7.1. The user starts at the position marked with the circle heading to the top, and walks along the track for four laps, ending at the starting position. The SNR averaged over the track is 14.9 dB.

The KEST results for that track are depicted in Fig. 7.8. The user is constantly in LoS to the physical transmitter. The LoS signal and the MPC starting with a propagation distance of approx. 97 m at the beginning are the only signal components tracked by KEST throughout the user trajectory. While other signal components seem to be tracked constantly, they are interrupted at some points.

The MAE for the user track in Fig. 7.7 is plotted in Fig. 7.9 in red for the case that no data association scheme is applied. For the curves in blue and orange, the data association scheme derived in Section 4.3 is applied with the ML data association method and data association sampling, respectively. No visibility mapping is performed. The red curve thus corresponds to the standard

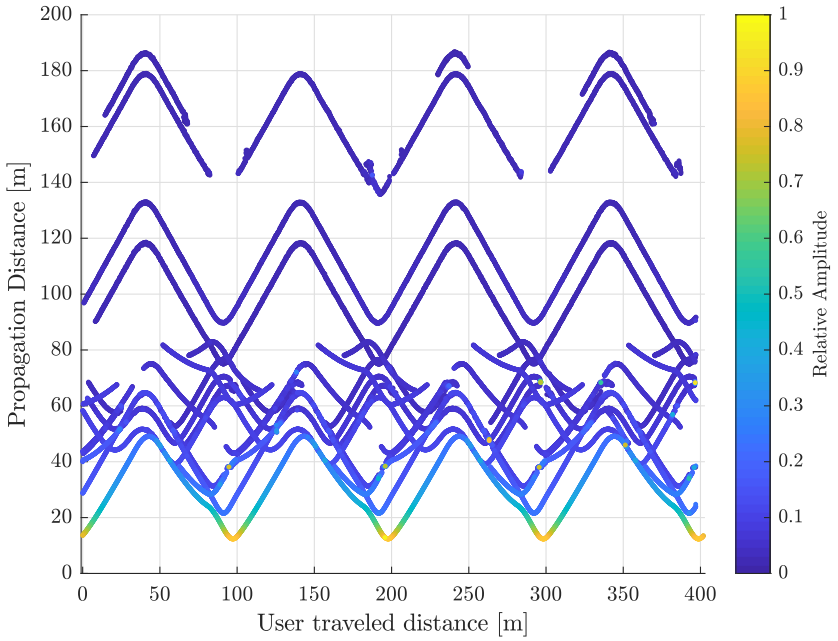


Figure 7.8: The KEST results for the data association track in Fig. 7.7. Every approx. 100 m, the user returns to the starting position and starts into a new loop.

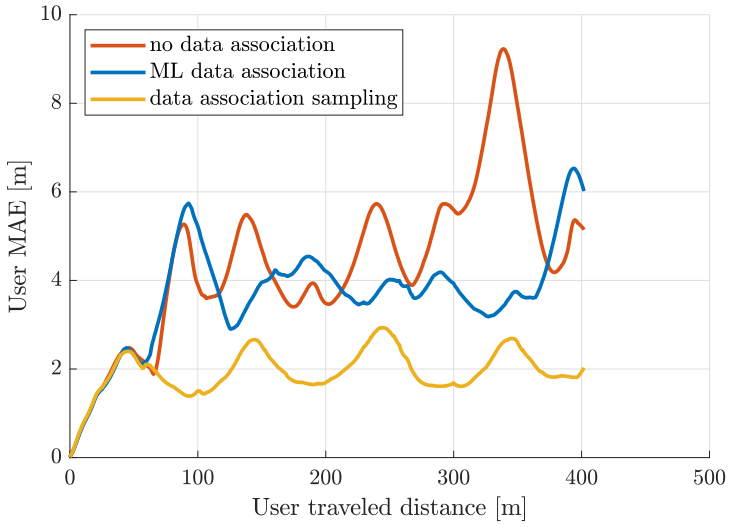


Figure 7.9: The MAE is plotted for the user track in Fig. 7.7. The red curve is the MAE with no data association scheme, the blue curve with ML data association, and the orange curve with data association sampling.

single user Channel-SLAM algorithm in Chapter 3.

Since the physical transmitter is in LoS throughout the user trajectory, the MAE for the red curve is bounded. Nevertheless, the error spikes periodically, as newly detected transmitters are initialized with a relatively high uncertainty.

For the orange curve corresponding to data association sampling, associations among the tracked signal components in Fig. 7.8 can be established. When a new signal component is detected, it can be associated with the corresponding transmitter that had been visible before. Consequently, new transmitters are initialized with a small uncertainty. The MAE is in this case considerably smaller than without a data association scheme, and it settles down at approx. 2 m.

With the ML data association scheme, the positioning performance is in the order of the performance without data association. Along the track, data association is often ambiguous. The ML method always decides for the association with highest likelihood as in Eq. (4.49). If this association decision is wrong, the respective user particles quickly become strongly biased towards the true user position. When a majority of user particles take wrong associations decisions, high errors in the user position estimate occur. In the resampling step of the particle filter, such biased particles are likely to not be resampled if they drift too far away from the true user state, and the user position estimate improves again. In this way, the spikes in the positioning performance in Fig. 7.9 can be explained for the blue curve regarding the ML association scheme.

In contrast, for data association sampling, the association decisions are not

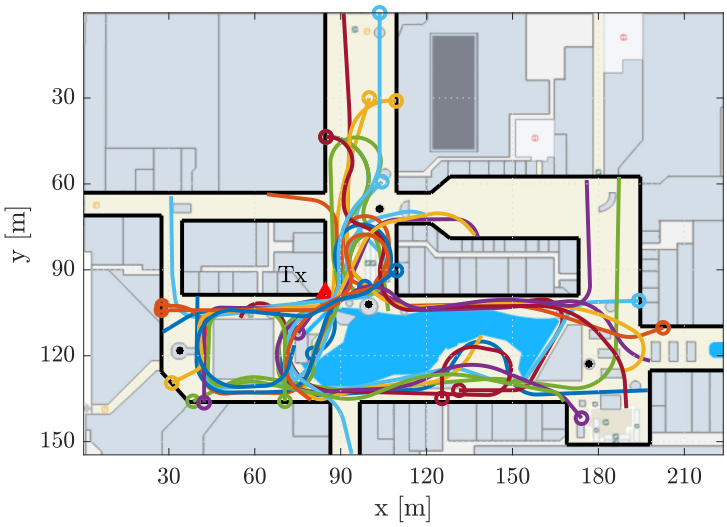


Figure 7.10: The simulation scenario with the user tracks to evaluate cooperative Channel-SLAM. The starting points of the tracks are marked by circles.

taken based on the highest likelihood, but sampled randomly based on the likelihoods. Thus, association decisions are distributed broader among the user particles, leading to a higher robustness and a better positioning performance.

7.3.4 Cooperative Channel-SLAM

This section evaluates the cooperative Channel-SLAM approach developed in the thesis. First, a user walks along the reference track in Fig. 7.1 doing Channel-SLAM without any prior information on the transmitters or the scenario. The user creates a transmitter map that is handed over to a second user who walks on a different trajectory through the scenario. The second user uses and updates the map as described in Chapter 5, and shares the map with the third user, and so on. In our evaluations, there are 21 users that successively receive, use and update the map originally created by the first user, following the flow chart in Fig. 5.6. They all walk along different tracks through the environment, which are plotted in Fig. 7.10. The start points of the tracks are marked by circles. The lengths and averaged SNRs at the receivers of these 21 tracks are summarized in Table 7.1. Including the first user walking along the reference track, 22 users have contributed to the final transmitter map.

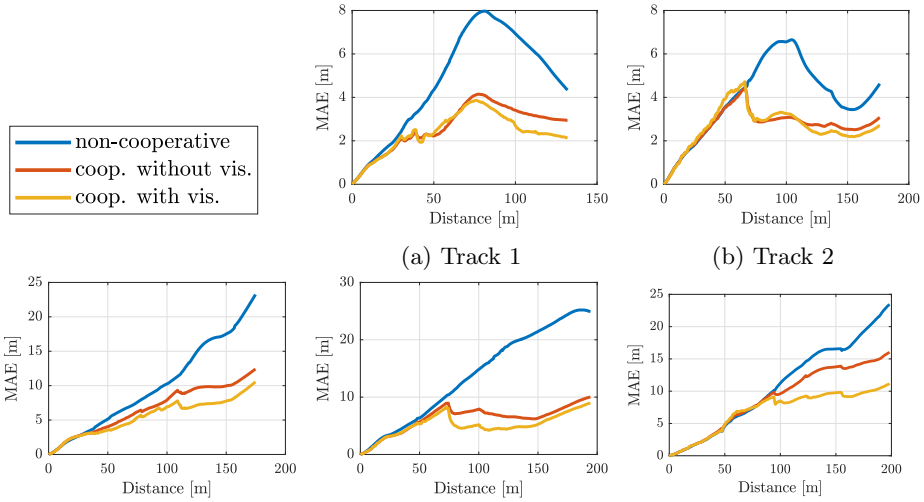
Fig. 7.11 exemplarily shows the user MAE for five of the involved tracks. The blue curves donate the MAE for single user, i.e., non-cooperative, Channel-SLAM. The data association scheme in Section 4.3, but no visibility mapping

Table 7.1: Average SNRs and lengths of the user tracks.

Track	1	2	3	4	5	6	7
SNR [dB]	15.52	17.59	14.12	10.80	11.61	13.22	10.42
Length [m]	131.9	176.3	143.2	210.7	204.6	174.7	176.8

Track	8	9	10	11	12	13	14
SNR [dB]	15.31	14.99	10.07	12.12	15.69	15.84	11.31
Length [m]	162.3	169.5	194.3	218.8	338.2	244.0	162.3

Track	15	16	17	18	19	20	21
SNR [dB]	17.01	10.24	13.84	12.16	7.82	9.35	10.07
Length [m]	81.2	163.6	176.4	138.4	217.8	198.3	222.7



(a) Track 1 (b) Track 2

(c) Track 6 (d) Track 10 (e) Track 20

Figure 7.11: The MAEs are plotted for five different tracks versus the user traveled distances in single user Channel-SLAM in blue and in cooperative Channel-SLAM without and with visibility information in red and orange, respectively.

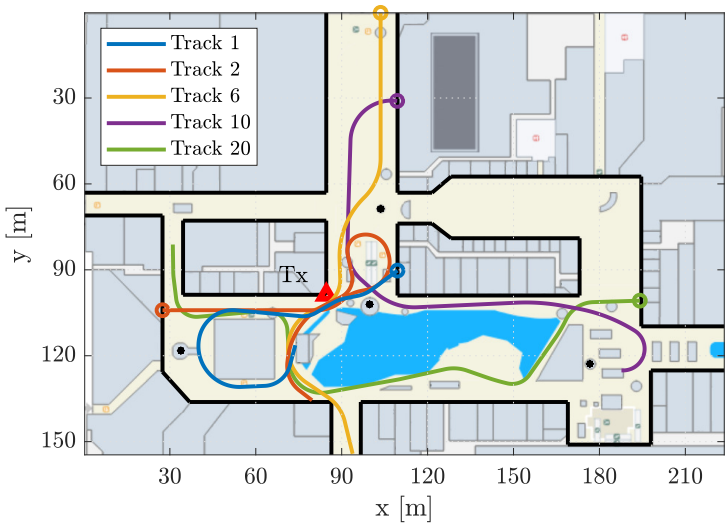


Figure 7.12: The simulation scenario with the user tracks from Fig. 7.10 that are evaluated in Fig. 7.11 are shown again. The starting positions of the users are marked by circles.

is applied.

The red curves are the MAE for cooperative Channel-SLAM as in Chapter 5 without visibility mapping. The transmitter map that is continuously used and updated contains estimates of transmitter states, but no visibility information.

The orange curves donate the MAE for cooperative Channel-SLAM with mapping and exploiting visibility information. Accordingly, the map that is used and updated by the users contains both estimated states and visibility information of transmitters. In this case, visibility information is mapped following Section 4.2 and used for data association following Section 4.3 and map matching following Section 5.2.7.

The user tracks from Fig. 7.10 that are evaluated in Fig. 7.11 are again plotted in Fig. 7.12 for clarity, where the circles mark the users' starting positions.

In the beginning, all three error curves essentially coincide in every plot in Fig. 7.11. They start from zero, since the initial position in the local coordinate system is known. Once a map match has been obtained, the respective prior map can be exploited in the cooperative Channel-SLAM cases, and associations among newly detected signal components and transmitters in the prior map can be established. The corresponding MAE is then considerably below the MAE for single user Channel-SLAM.

The tracks in Fig. 7.11 vary in their positioning performances. For example,

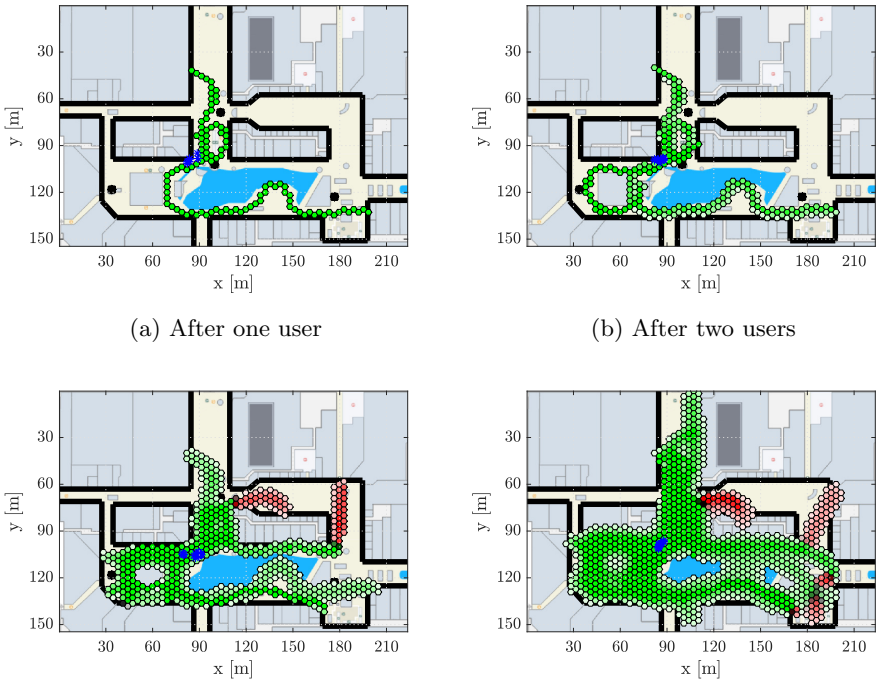
the MAEs of the tracks in Figs. 7.11 (a) and 7.11 (b) are never greater than 8 m. In contrast, the errors for the tracks in Figs. 7.11 (c), 7.11 (d) and 7.11 (e) are in the non-cooperative case in the order of 25 m near the end of the tracks. From Fig. 7.11 can be seen that the tracks in Figs. 7.11 (a) and 7.11 (b) start relatively close to the physical transmitter. Due to a favorable geometrical dilution of precision situation, the initial uncertainty about the physical transmitter's and some virtual transmitters' states shrinks quickly for these two tracks. Since in addition the physical transmitter is always in LoS and tracked by KEST throughout the tracks, the positioning performance is relatively good. The tracks in Figs. 7.11 (c), 7.11 (d) and 7.11 (e) start at locations with unfavorable geometrical dilutions of precision. The high uncertainty about the transmitter states shrinks only slowly, leading to a high uncertainty about the user location. The geometrical dilutions of precision, in particular at the initial user position, has a considerable impact on the positioning performance of both single user and cooperative Channel-SLAM.

The information regarding the visibility of transmitters affects the positioning performance of the tracks in Fig. 7.11 differently as well. For example, visibility information has not much influence on the positioning performance in 7.11 (b), while the positioning performance is significantly better in 7.11 (e) if visibility information is estimated and exploited. As mentioned in Section 4.3 and Section 5.2.7, visibility information increases the robustness of data association and map matching mostly in case of ambiguities. Depending on the user track and the transmitters in the prior map, association and map matching ambiguities are more or less likely, and the influence of visibility information is accordingly significant or small.

In particular, if there are many transmitters mapped by the user and only few transmitters in the prior transmitter map or vice versa, such ambiguities are more likely and the influence of visibility higher. In addition, high uncertainties in transmitter state estimates in the user and prior map cause these ambiguities. In contrast, if the transmitter state estimates in the prior map and in the user map have converged, ambiguities are less likely, and the influence of visibility information on the positioning performance is relatively small.

In short, visibility information significantly contributes to the positioning performance in three cases. The first case is the detection of loop closures. If a user returns to a previously visited hexagon, the mapped visibility information helps in estimating the user location in the particle filter derived in Section 4.2. In the second case, visibility information improves the robustness of the data association scheme and of map matching if only few transmitters are in the prior map, i.e., if the map learning process is in an early stage and only few users have contributed to the prior map. In the third case, the user is in the beginning of their track and has mapped few transmitters with a yet relatively high uncertainty. Again, visibility information contributes to the robustness of the data association scheme and of map matching.

Fig. 7.13 exemplarily shows the evolution of the hexagonal visibility map for the physical transmitter after one, two, six and twelve users have contributed to the map. Green hexagons indicate that the transmitter is visible, while red



(a) After one user

(b) After two users

(c) After six users

(d) After twelve users

Figure 7.13: The visibility maps for the physical transmitter are plotted exemplarily after the contribution of one, two, six, and twelve users. Green hexagons indicate that the transmitter is visible, and red hexagons that it is not. The opacity indicates the reliability of the visibility estimates.

hexagons indicate that it is not. The opacity of the hexagon colors corresponds to the variance of the respective Beta distribution. A high opacity thus corresponds to a high certainty about the transmitter visibility. The transmitter location estimate is plotted by the dark blue particles.

The evolution of the number of transmitters and hexagons in the prior map in cooperative Channel-SLAM is plotted in Fig. 7.14. Both curves grow linearly in the beginning and start to flatten out towards the end.

After a transmitter map was created, shared, used and updated by the previous 22 users, a final user walks along the reference track again with this transmitter map as prior knowledge.

Fig. 7.15 shows the MAE curve for this final user on the reference track for three cases. The first case drawn blue is single user Channel-SLAM with data association, but without visibility information, and corresponds to the error curve in Fig. 7.4. The second case corresponding to the red curve is the MAE of the final user with the prior transmitter map created by the 22

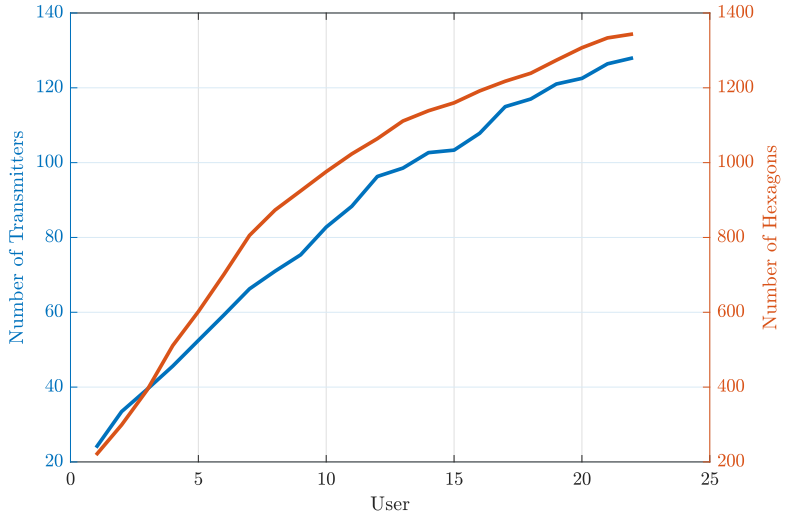


Figure 7.14: The number of transmitters and the number of hexagons in the prior map in cooperative Channel-SLAM are plotted versus the number of cooperating users.

users. In this second case, no visibility information was mapped or used in the data association and map matching schemes. Finally, the third case with the orange MAE curve corresponds to the second case, but all users map and exploit visibility information for both data association and map matching. The MAE curves for the cooperative cases are bounded in the order of $3 - 4\,\mathrm{m}$ and throughout most of the track considerably lower than the MAE for the single user. Only near the end of the track, the errors start to increase slightly. This effect can be explained with the bad geometrical dilution of precision in this region. All signal components arrive at the user from a similar direction, which introduces an uncertainty into the user location estimate.

The positioning performance with and without visibility information are almost identical. As mentioned above, visibility information increases the robustness of data association and map matching if ambiguities arise. Many of the transmitters visible from the reference track have been mapped in the prior map by the previous users with a relatively low uncertainty about their state. Data association and map matching ambiguities are therefore unlikely, and visibility information has hardly any influence on the positioning performance along the reference track.

Fig. 7.16 (a) shows the processing time of map matching for the single time instants versus the user traveled distance for the reference track. A zoom into the part of the figure framed by the dotted rectangle is provided in Fig. 7.16 (b). It can be seen that map matching is performed every 2 m of traveled distance. As mentioned in Section 5.2, map matching is computationally expensive, especially

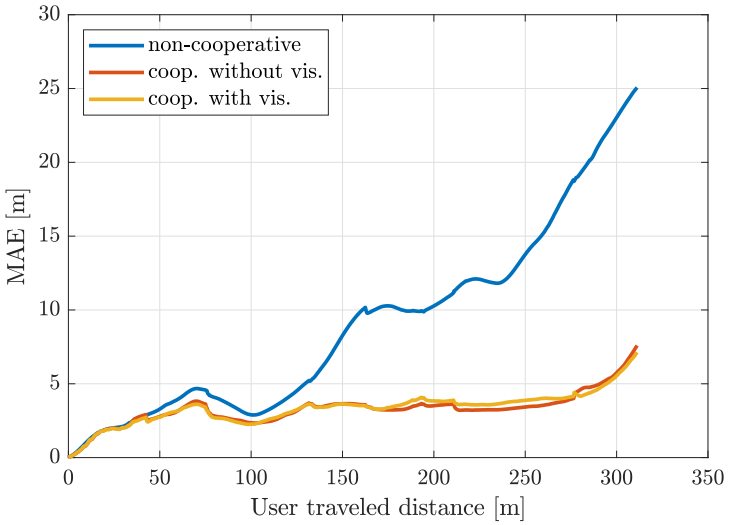


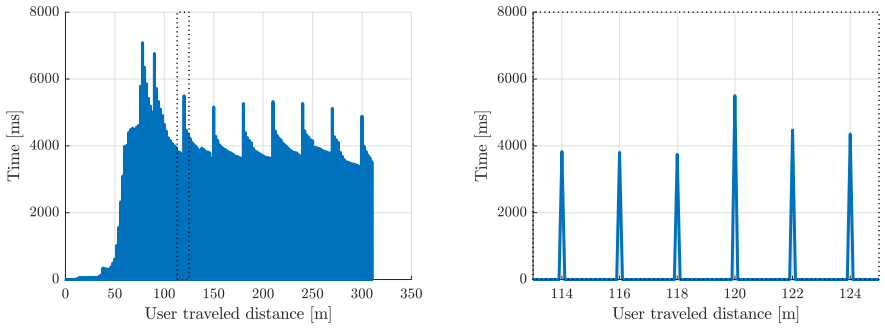
Figure 7.15: The MAE is plotted for the user track in Fig. 7.1 for single user Channel-SLAM in blue, and for cooperative Channel-SLAM without and with visibility information in red and orange, respectively. In the cooperative cases, a prior transmitter map that has been created by the 22 users with tracks depicted in Fig. 7.1 and Fig. 7.10 is used.

if many transmitters are involved. However, it is not performed at every time instant, and it can be performed in parallel to the Channel-SLAM algorithm.

To evaluate the convergence of the transmitter map, we regard again a cooperative scenario, where 21 different users walk along the tracks in Fig. 7.10. The first user starts with no prior knowledge, i.e., with no prior transmitter map, creates such a map with Channel-SLAM and hands it on to the second user. All other 20 users receive the transmitter map from the respective previous user, use it as they travel along their track, update it and pass it on to the respective next user in the cooperative Channel-SLAM framework.

Each time the map is passed from one user to another, it is also passed on to a reference user who walks along the reference track in Fig. 7.1 with that map as prior information. The results in terms of MAE of the reference user exploiting the prior map after one, two, six, ten, 15 and 21 users have contributed to the map are plotted in Fig. 7.17. In addition, the blue curve in Fig. 7.17 refers to single user Channel-SLAM as reference.

From Fig. 7.17, it becomes evident that a transmitter map that has been created by one single user in the same scenario improves the positioning performance considerably, if multiple transmitters are mapped already with a relatively small uncertainty about their state. If an additional user has contributed to the prior map, the positioning performance of the reference user improves again substantially near the end of the reference track. In this area, fewer



(a) Processing time across the track

(b) Processing time zoomed

Figure 7.16: The processing times for map matching along the reference track are plotted versus the user traveled distance. In (a), the processing times across the entire track are shown. In (b), the same plot is shown zoomed into the dotted rectangle in (a).

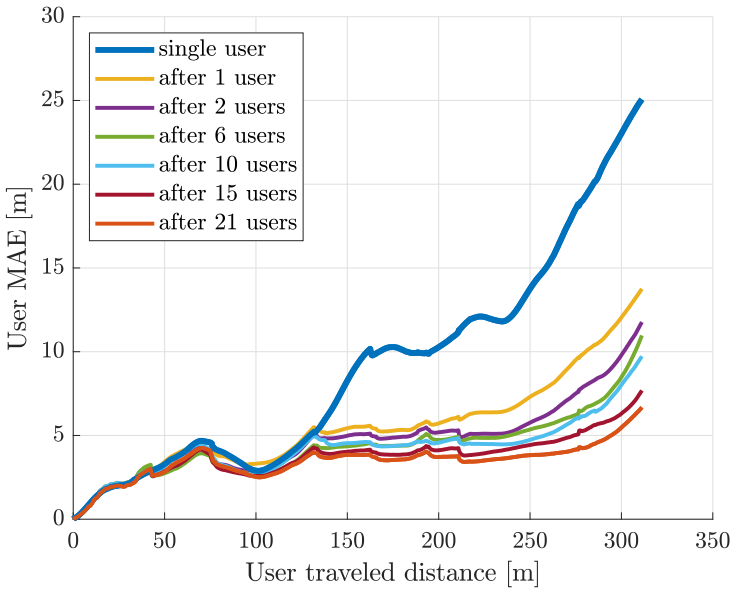


Figure 7.17: The MAEs are plotted for users walking along the reference track in Fig. 7.1. The single user Channel-SLAM case is depicted in blue. The other curves show the MAE of a user walking along the reference track with cooperative Channel-SLAM after a number of users have contributed to the prior transmitter map. These users have traveled along the respective tracks in Fig. 7.10.

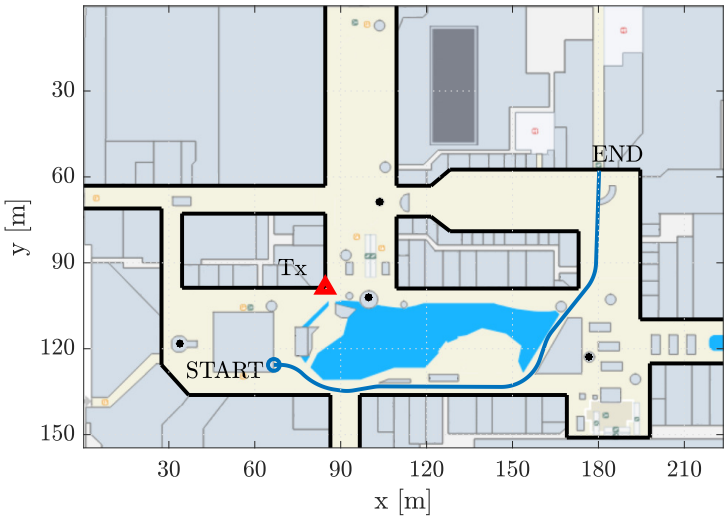


Figure 7.18: The simulation scenario with the user track for the GSPF evaluation.

transmitters have been mapped with a converged state estimate, and the geometrical dilution of precision is relatively high. Consequently, more and better transmitter state and visibility estimates in the prior map can improve the positioning performance in particular in such areas with few mapped transmitters and a high geometrical dilution of precision.

7.3.5 The Gaussian Sum Particle Filter

The track for evaluating the GSPF from Chapter 6 is plotted in Fig. 7.18. The KEST results for the track in Fig. 7.18 are plotted in Fig. 7.19. From Fig. 7.19 can be observed that there are at least three of the transmitters visible at any user position. The LoS component corresponds to the signal component with lowest propagation distance, which is tracked from the beginning to a traveled distance of 129.1 m. Afterward, during the last approx. 40 m of the track, the user is in a NLoS condition and makes only use of virtual transmitters for localization.

The MAE of single user Channel-SLAM with data association is plotted in Fig. 7.19 for the track in Fig. 7.18 for the two cases. In the first case corresponding to the blue line, the GSPF as in Chapter 6 is used, where the transmitter states are represented by Gaussian mixture models and estimated by unscented Kalman filters. For the second case drawn red, the Channel-SLAM algorithm as derived in Chapter 4 is used, where the transmitter states are estimated using

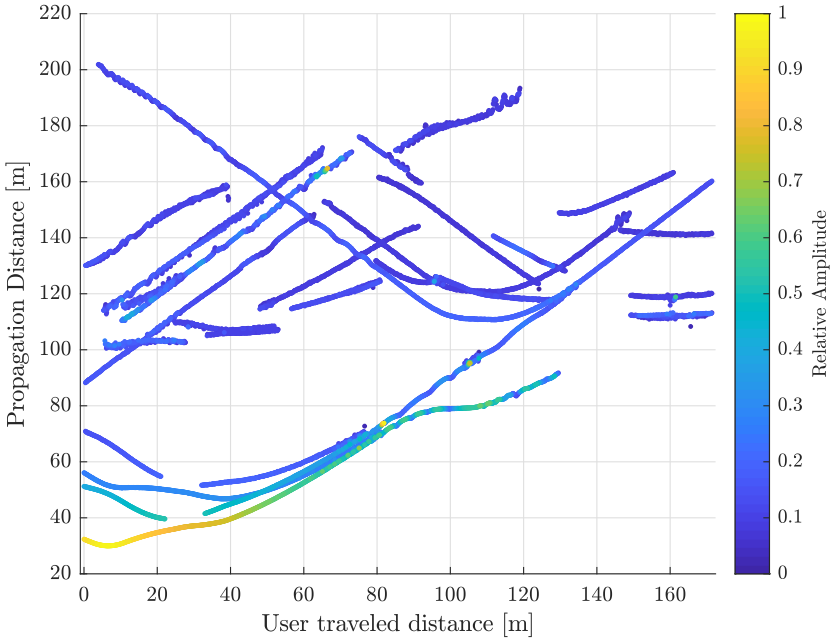


Figure 7.19: The KEST results for the user track in Fig. 7.18. The LoS component is tracked from the beginning to a traveled distance of 129.1 m.

particle filters.

The performance for both implementations is almost the same, which is true for all user tracks in Fig. 7.10. Since track of transmitters is lost and new transmitters show up constantly, the data association scheme can hardly contribute to the positioning performance, and the MAE keeps increasing over time.

The processing times of the Channel-SLAM algorithm are plotted in Fig. 7.1 for the implementation of the transmitter states estimated by unscented Kalman filters as in Chapter 6 and by particle filters as in Chapter 4. In both cases, there are 500 user particles used. While the processing time for the unscented Kalman filter case is much higher, it can still be decreased with a more efficient implementation of numerical calculations in C++, such as matrix inversions and decompositions. A black line is drawn at a time of 100 ms, indicating the time between time instants in Channel-SLAM. If the processing time curves are below the black line, the filter runs in real-time. Thus, the GSPF for Channel-SLAM does not run in real-time throughout the track, while the Rao-Blackwellized particle filter runs in real-time for only 13.5 m of traveled distance.

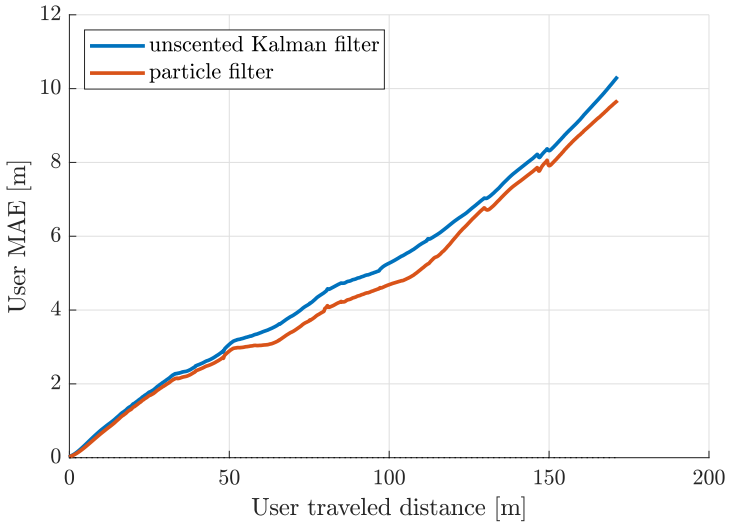


Figure 7.20: The MAE is plotted in blue for the user track in Fig. 7.18 versus the distance the user has traveled. The curve in blue refers to the GSPF in Chapter 6, and the curve in red to the Channel-SLAM filter in Chapter 4.

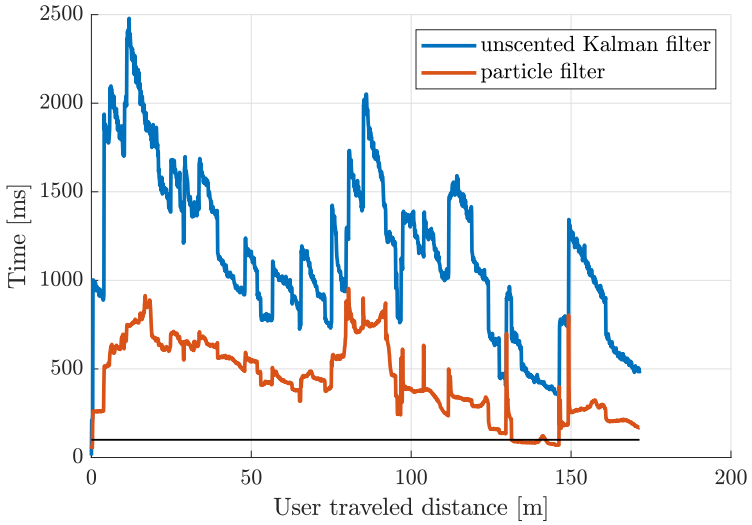


Figure 7.21: The processing times of Channel-SLAM are plotted for the user track in Fig. 7.18. The curve in blue refers to the GSPF in Chapter 6, and the curve in red to the Channel-SLAM filter in Chapter 4.

Chapter 8

Conclusion

Multipath assisted positioning exploits multipath propagation for positioning by considering MPCs as LoS signals from virtual transmitters. Channel-SLAM is a multipath assisted positioning algorithm that simultaneously estimates the position of the user and maps the states of physical and virtual transmitters. Though, Channel-SLAM has two major limitations, namely the limitation to a single user and the lack of a data association method.

Data association is a crucial element for any long term robust SLAM scheme, as it finds correspondences among landmarks. Without data association, the uncertainty about the user position increases and the positioning error accumulates over time.

In single user positioning schemes, users estimate their position on their own. The high uncertainty about transmitter states upon initialization translates into a limited positioning performance of the user in Channel-SLAM. The enormous potential of cooperation among user is unused.

In this thesis, the existing Channel-SLAM algorithm has been extended by a reliable data association scheme and cooperation. In both data association and cooperation by transmitter map exchange, it is crucial to be able to identify transmitters and differentiate among them. Hence, the diversity among transmitters needs to be sufficiently high. However, in multipath assisted positioning algorithms, this diversity among transmitters is very poor, since virtual transmitters correspond only to reflected and scattered versions of the transmit signal.

To increase the diversity among transmitters, we have introduced the concept of visibility in Chapter 4. The idea is to differentiate among transmitters not only based on their states, but also based on their local visibilities. To gain such visibility information, we have derived a new particle filter that both estimates and exploits information on transmitter visibilities. In particular, visibility information in Channel-SLAM facilitates detecting loop closures. When a user comes back to a previously visited location, previously mapped transmitters are expected to be visible again.

The challenge of data association in multipath assisted positioning is to

associate signal components with transmitters. The increased diversity among transmitters when considering visibility areas facilitates data association considerably by resolving possible association ambiguities. With the multiple hypothesis tracking data association scheme for Channel-SLAM derived in Chapter 4, the error of a user's position estimate can be bounded in the long term. Due to the robustness of the data association scheme coming from the increased diversity among transmitters, its complexity in terms of additional user particles that are necessary is negligible.

Since Channel-SLAM is a relative positioning approach, the coordinate system of a user is random and not aligned with the coordinate system of a prior map. The transformation parameters relating the coordinate systems are in general unknown. When maps are shared, map matching is the fundamental problem, as both the transformation parameters and correspondences among transmitters need to be estimated. We have presented a map matching scheme in Chapter 5, which is based on the RANSAC algorithm. Certain conditions such as a low number of transmitters observed by the user or in the prior map, or scenarios with specific symmetries can lead to ambiguities in map matching. Such ambiguities in turn can lead to heavily biased estimates of the transformation parameters and transmitter correspondences. Thus, visibility information about transmitters is incorporated in our map matching scheme by a likelihood ratio test to resolve these ambiguities. The robustness of the map matching scheme enables the creation and constant improvement of transmitter maps with crowdsourcing. If the relation of a transmitter map to a global coordinate system is known, Channel-SLAM turns from a relative to an absolute localization system, marking a fundamental shift in the concept of Channel-SLAM.

When maps are exchanged among users, information about transmitter states and visibilities is transmitted. Mapping the transmitters' states may require a high amount of memory, since they are represented as particle clouds in the Rao-Blackwellized particle filter in Channel-SLAM. One option to avoid a high amount of data to be transferred is to approximate the particle clouds with parametrized distributions that are represented with few parameters before a map is shared. A different approach to decrease the amount of data has been taken in Chapter 6, where a new tracking filter named Gaussian sum particle filter has been derived for Channel-SLAM. It represents the transmitter states by Gaussian mixture models and estimates the single Gaussian components by unscented Kalman filters.

The methods developed in this thesis have been evaluated in Chapter 7. As expected, the data association scheme can significantly improve the positioning performance of single user Channel-SLAM, if the user revisits locations where they had been earlier. Also, data association helps if track of signals components is lost temporarily in KEST, for example due to blocking or if two or more signal components can not be resolved. Nevertheless, when the user track is such that only few or no associations can be established, the data association scheme can not contribute to a better positioning performance.

In this case, cooperation can significantly improve the positioning performance. With crowdsourcing, maps of transmitter states and visibilities are

created. A user who has obtained such a map can associate newly detected signal components with transmitters from the prior map. The initial uncertainty about the transmitters' states and visibilities is often drastically decreased, leading to a significant decrease of the uncertainty about the user position. If the prior map has converged and the geometrical dilution of precision is low, the positioning error of a user is bounded in the long term.

Visibility information can increase the robustness of data association and map matching considerably. In particular during the learning phase of a transmitter map of a scenario, such information can therefore improve the user positioning estimate and in turn the transmitter state and visibility estimates in the map. Likewise, in the beginning of a user track when a user has to rely on only few converged transmitter state estimates, visibility information contributes significantly to the robustness of data association and map matching.

Future research topics in multipath assisted positioning and Channel-SLAM could include the following.

- Channel-SLAM is a two step approach. In a first step, the parameters of signal components are estimated, while in a second step, these estimates are used to estimate the user position and map the transmitter states. Combining these two steps into one can bring additional gain in terms of robustness and accuracy.

- The Channel-SLAM algorithm considers two-dimensional scenarios. In reality, it is unlikely that transmitters and receivers are all in a two-dimensional plane. For example, ground reflections or scatterers on the ceiling are not considered by Channel-SLAM and may introduce biases. The extension to three-dimensional scenarios is therefore worth investigating.

- The cooperation in this thesis refers to sharing of transmitter maps. A different form of cooperation in Channel-SLAM is ranging among user transceivers, when users move in a scenario at the same time. Such range estimates among a network of users bring further benefit, and they could be integrated into the current Channel-SLAM algorithm. In that sense, a decentralized estimation of multiple users' states and transmitter states would be performed.

- Cooperative Channel-SLAM in its current implementation does not run in real-time in most cases. Both steps of Channel-SLAM, the channel parameter estimation and the state and visibility estimation can be improved in terms of complexity in both an algorithmic and an implementational sense. Combining the channel parameter estimation with the user and transmitter state and visibility estimation in a single step may not only improve the positioning performance, but also reduce the computational complexity considerably. Especially the high processing times of the channel estimator KEST may be avoided in this way.

Appendix A

Kalman Filter Equations

The following sections recap the equations for the Kalman filter and the unscented Kalman filter. The state estimate at time instant k is denoted by $\boldsymbol{x}_k$, and its covariance matrix estimate by $\boldsymbol{P}_k$. The noise samples $\boldsymbol{w}_k$ and $\boldsymbol{n}_k$ are assumed to be drawn drawn from normal distributions with zero mean and covariance matrices $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$, respectively.

A.1 Standard Kalman Filter

The Kalman filter assumes the system model at time instant k to be of the form

$$\boldsymbol{x}_k = \boldsymbol{A}_k \boldsymbol{x}_{k-1} + \boldsymbol{B}_k \boldsymbol{u}_k + \boldsymbol{w}_k, \tag{A.1}$$

where $\boldsymbol{A}_k$ models the state transition, $\boldsymbol{u}_k$ is the control input and $\boldsymbol{B}_k$ its model. The noise sample $\boldsymbol{w}_k$ is drawn from a zero-mean Gaussian distribution of known covariance matrix $\boldsymbol{Q}_k$. The measurement model is assumed to be of the form

$$\boldsymbol{z}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{n}_k, \tag{A.2}$$

where $\boldsymbol{H}_k$ is the measurement model relating the state estimate $\boldsymbol{x}_k$ to the measurement $\boldsymbol{z}_k$. The noise sample $\boldsymbol{n}_k$ is drawn from a zero-mean Gaussian distribution of known covariance matrix $\boldsymbol{R}_k$.

A.1.1 Prediction

The prediction in the Kalman filter involves the prediction of the state estimate,

$$\boldsymbol{x}_k^- = \boldsymbol{A}_k \boldsymbol{x}_{k-1} + \boldsymbol{B}_k \boldsymbol{u}_k, \tag{A.3}$$

and the prediction of the covariance matrix estimate,

$$\boldsymbol{P}_k^- = \boldsymbol{A}_k \boldsymbol{P}_{k-1} \boldsymbol{A}_k^T + \boldsymbol{Q}_k. \tag{A.4}$$

A.1.2 Update

In the update step, first the Kalman gain $\boldsymbol{K}_k$ is calculated by

$$\boldsymbol{K}_k = \boldsymbol{P}_k^- \boldsymbol{H}_k^T \left(\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{R}_k \right)^{-1}. \tag{A.5}$$

Afterward, the state estimate is updated with the measurement by

$$\boldsymbol{x}_k = \boldsymbol{x}_k^- + \boldsymbol{K}_k \left(\boldsymbol{z}_k - \boldsymbol{H}_k \boldsymbol{x}_k^- \right). \tag{A.6}$$

Finally, the covariance matrix is calculated as

$$\boldsymbol{P}_k = \left(\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k \right) \boldsymbol{P}_k^-. \tag{A.7}$$

A.2 Unscented Kalman Filter

The system and measurement model are given by Eq. (2.12) and Eq. (2.13), respectively.

A.2.1 Prediction

Given $\boldsymbol{x}_{k-1}$ and $\boldsymbol{P}_{k-1}$, first the N_{sig} sigma points $\boldsymbol{X}_{m,k-1}$ and their weights $\omega_{m,k-1}$ are calculated for $m = 1, \ldots, N_{\text{sig}}$, for example by Eq. (2.19). The sigma points are propagated through the system model as

$$\boldsymbol{X}_{m,k-1}^- = \boldsymbol{f}_k \left(\boldsymbol{X}_{m,k-1}, \boldsymbol{u}_k \right) \tag{A.8}$$

for $m = 1, \ldots, N_{\text{sig}}$. The prediction of the state estimate is calculated as

$$\boldsymbol{x}_k^- = \sum_{m=1}^{N_{\text{sig}}} \omega_{m,k-1} \boldsymbol{X}_{m,k-1}^-, \tag{A.9}$$

and the prediction of the covariance matrix estimate as

$$\boldsymbol{P}_k^- = \sum_{m=1}^{N_{\text{sig}}} \omega_{m,k-1} \boldsymbol{X}_{m,k-1}^- {\boldsymbol{X}_{m,k-1}^-}^T - \boldsymbol{x}_k^- {\boldsymbol{x}_k^-}^T + \boldsymbol{Q}_k. \tag{A.10}$$

A.2.2 Update

Given $\boldsymbol{x}_k^-$ and $\boldsymbol{P}_k^-$, the N_{sig} sigma points $\boldsymbol{X}_{m,k}^-$ and their weights $\omega_{m,k}$ are calculated for $m = 1, \ldots, N_{\text{sig}}$. The sigma points are propagated through the measurement model as

$$\boldsymbol{Z}_{m,k}^- = \boldsymbol{h}_k \left(\boldsymbol{X}_{m,k}^- \right) \tag{A.11}$$

for $m = 1, \ldots, N_{\text{sig}}$. The prediction of the measurement is calculated by

$$\boldsymbol{z}_k^- = \sum_{m=1}^{N_{\text{sig}}} \omega_{m,k} \boldsymbol{Z}_{m,k}^-. \tag{A.12}$$

The innovation covariance matrix

$$\boldsymbol{P}_{zz,k} = \sum_{m=1}^{N_{\text{sig}}} \omega_{m,k} \boldsymbol{Z}_{m,k}^{-} \boldsymbol{Z}_{m,k}^{-}{}^{T} - \boldsymbol{z}_{k}^{-} \boldsymbol{z}_{k}^{-}{}^{T} + \boldsymbol{R}_{k} \tag{A.13}$$

and the cross-covariance matrix

$$\boldsymbol{P}_{xz,k} = \sum_{m=1}^{N_{\text{sig}}} \omega_{m,k} \boldsymbol{X}_{m,k}^{-} \boldsymbol{Z}_{m,k}^{-}{}^{T} - \boldsymbol{x}_{k}^{-} \boldsymbol{z}_{k}^{-}{}^{T} \tag{A.14}$$

are used to compute the Kalman gain

$$\boldsymbol{K}_{k} = \boldsymbol{P}_{xz,k} \boldsymbol{P}_{zz,k}{}^{-1}. \tag{A.15}$$

Finally, the updated state estimate and covariance matrix are

$$\boldsymbol{x}_{k} = \boldsymbol{x}_{k}^{-} + \boldsymbol{K}_{k} \left(\boldsymbol{z}_{k} - \boldsymbol{z}_{k}^{-} \right) \tag{A.16}$$

and

$$\boldsymbol{P}_{k} = \boldsymbol{P}_{k}^{-} - \boldsymbol{K}_{k} \boldsymbol{P}_{zz,k} \boldsymbol{K}_{k}{}^{T}, \tag{A.17}$$

respectively.

Appendix B

Derivations for the Particle Filter with Visibility Information

In the following two sections, more details on the derivation of the particle filter with visibility information from Section 4.2 are presented.

B.1 Weight Derivation

This section derives Eq. (4.18) from Eq. (4.15). With Bayes' theorem, the numerator in Eq. (4.15) can generally be written as

$$\begin{aligned} \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) = & \frac{\mathrm{p}\left(\boldsymbol{z}_k | \boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k}, \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right)}{\mathrm{p}\left(\boldsymbol{z}_k | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right)} \\ & \times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right). \end{aligned} \tag{B.1}$$

The user state $\boldsymbol{x}_{\mathrm{u},k}$ at time instant k does not depend on measurements and control inputs at time instants before $k-1$ if $\boldsymbol{x}_{\mathrm{u},k-1}$ is known. In addition, information from future time instants is not incorporated. Hence, the factor in the second line on the right hand side in Eq. (B.1) is further decomposed by

$$\begin{aligned} \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right) &= \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}, \boldsymbol{M}_k | \boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{M}_{0:k-1}, \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right) \\ &\quad \times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{M}_{0:k-1} | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k-1}\right) \\ &= \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}, \boldsymbol{M}_k | \boldsymbol{x}_{\mathrm{u},k-1}, \boldsymbol{M}_{k-1}, \boldsymbol{u}_k\right) \\ &\quad \times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{M}_{0:k-1} | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k-1}\right). \end{aligned} \tag{B.2}$$

The denominator in the first line on the right hand side of Eq. (B.1) does not depend on $\boldsymbol{x}_{\mathrm{u}}$ or $\boldsymbol{M}$ and is therefore a constant. Inserting Eq. (B.2) into

Eq. (B.1) yields

$$\begin{aligned}\mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \propto\ & \mathrm{p}\left(\boldsymbol{z}_k | \boldsymbol{x}_{\mathrm{u},0:k}, \boldsymbol{M}_{0:k}, \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right) \\ & \times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},k}, \boldsymbol{M}_k | \boldsymbol{x}_{\mathrm{u},k-1}, \boldsymbol{M}_{k-1}, \boldsymbol{u}_k\right) \\ & \times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k-1}, \boldsymbol{M}_{0:k-1} | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k-1}\right).\end{aligned} \tag{B.3}$$

The weights in Eq. (4.18) result from inserting Eq. (B.3) and Eq. (4.16) into Eq. (4.15).

B.2 Radio Measurement Likelihood

In the following, the likelihood in Eq. (4.33) is derived. The control input at time instant k has no information on the radio measurement if the user state is known at that time instant. Marginalizing over the transmitters' state yields

$$\begin{aligned}&\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}, \boldsymbol{u}_{1:k}\right) \\ &\quad = \prod_{j=1}^{N_{\mathrm{TX},k}^{\mathrm{a}}} \int \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k} | \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}, \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}\right) \\ &\qquad\qquad \times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \mathrm{d}\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}.\end{aligned} \tag{B.4}$$

Since the radio measurement depends only on the user location, transmitter states and, for a correct association of the radio measurements with the transmitters, visibilities at time instant k, the first factor in the integral in Eq. (B.4) is

$$\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k} | \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}, \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}\right) = \mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k} | \boldsymbol{z}_{\mathrm{V},k}, \boldsymbol{x}_{\mathrm{u},k}^{<i>}, \boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}\right). \tag{B.5}$$

For the second factor in the integral in Eq. (B.4), marginalizing over the transmitter state at time instant $k-1$ yields

$$\begin{aligned}&\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\ &\quad = \int \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>} | \boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j>}, \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{\mathrm{V},1:k}, \boldsymbol{z}_{\mathrm{R},1:k-1}\right) \\ &\qquad \times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j>} | \boldsymbol{x}_{\mathrm{u},0:k}^{<i>}, \boldsymbol{z}_{1:k-1}\right) \mathrm{d}\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j>}\end{aligned} \tag{B.6}$$

$$\begin{aligned}&\quad = \int \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>} | \boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j>}\right) \\ &\qquad \times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j>} | \boldsymbol{x}_{\mathrm{u},0:k-1}^{<i>}, \boldsymbol{z}_{1:k-1}\right) \mathrm{d}\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j>}.\end{aligned} \tag{B.7}$$

The second factor in the integral in Eq. (B.7) is the posterior PDF of the state of the j^{th} transmitter at time instant $k-1$. It can be represented as a

number of $N_{p,\mathrm{Tx}}$ transmitter particles $\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j,\ell>}$ with weights $w_{k-1}^{<i,j,\ell>}$ analogue to Eq. (3.18). Thus, Eq. (B.7) can be written as

$$\begin{aligned}
&\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>},\boldsymbol{z}_{\mathrm{V},1:k},\boldsymbol{z}_{\mathrm{R},1:k-1}\right)\\
&\quad=\int\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}|\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j>}\right)\\
&\qquad\times\sum_{\ell=1}^{N_{p,\mathrm{Tx}}}w_{k-1}^{<i,j,\ell>}\delta\left(\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j>}-\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j,\ell>}\right)\mathrm{d}\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j>}
\end{aligned}\tag{B.8}$$

$$=\sum_{\ell=1}^{N_{p,\mathrm{Tx}}}w_{k-1}^{<i,j,\ell>}\,\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}|\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j,\ell>}\right),\tag{B.9}$$

where the very last term corresponds to the movement model of the transmitters as in Eq. (3.19). Inserting Eqs. (B.5) and (B.9) into Eq. (B.4) results in

$$\begin{aligned}
&\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}|\boldsymbol{x}_{\mathrm{u},0:k}^{<i>},\boldsymbol{z}_{\mathrm{V},1:k},\boldsymbol{z}_{\mathrm{R},1:k-1}\right)\\
&\quad=\prod_{j=1}^{N_{\mathrm{TX},k}^{\mathrm{a}}}\int\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{z}_{\mathrm{V},k},\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}\right)\\
&\qquad\times\sum_{\ell=1}^{N_{p,\mathrm{Tx}}}w_{k-1}^{<i,j,\ell>}\,\mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}|\boldsymbol{x}_{\mathrm{TX},k-1}^{<i,j,\ell>}\right)\mathrm{d}\boldsymbol{x}_{\mathrm{TX},k}^{<i,j>}
\end{aligned}\tag{B.10}$$

$$=\prod_{j=1}^{N_{\mathrm{TX},k}^{\mathrm{a}}}\sum_{\ell=1}^{N_{p,\mathrm{Tx}}}w_{k-1}^{<i,j,\ell>}\,\mathrm{p}\left(\boldsymbol{z}_{\mathrm{R},k}|\boldsymbol{z}_{\mathrm{V},k},\boldsymbol{x}_{\mathrm{u},k}^{<i>},\boldsymbol{x}_{\mathrm{TX},k}^{<i,j,\ell>}\right).$$

List of Important Expressions and Variables

$\lvert \cdot \rvert$	absolute value of a scalar or cardinality of a set
$\det(\cdot)$	determinant of a matrix
$\lVert \cdot \rVert$	Euclidean norm of a vector
$\mathbb{E}[\cdot]$	expectation value
$\mathrm{Tr}(\cdot)$	trace of a matrix
$(\cdot)^T$	transpose of a vector or matrix
$\mathrm{atan2}(x, y)$	four-quadrant inverse tangent function
$\mathcal{B}(x; p, q)$	PDF of the Beta distribution in x
$\mathrm{B}(p, q)$	Beta function
$\delta(\boldsymbol{x})$	Dirac delta distribution
$\Gamma(p)$	Gamma function
$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_{\boldsymbol{x}}, \boldsymbol{C}_{\boldsymbol{x}})$	Gaussian PDF in $\boldsymbol{x}$ with mean $\boldsymbol{\mu}_{\boldsymbol{x}}$ and covariance matrix $\boldsymbol{C}_{\boldsymbol{x}}$
$\mathbb{0}_n$	zero matrix of dimension $n \times n$
$\mathbb{1}_n$	identity matrix of dimension $n \times n$
h	index of a hexagon
i	index of a user or user-map particle
j	index of a signal component or transmitter
ℓ	index of a transmitter particle or Gaussian component
k	time instant
T_k	time between time instants $k-1$ and k

$\boldsymbol{x}^{<i>}$	i^{th} particle
$w^{<i>}$	weight of the i^{th} particle
$\boldsymbol{X}_m$	m^{th} sigma point
ω_m	weight of the m^{th} sigma point
$N_{\mathcal{N}}$	number of components in a Gaussian mixture
N_H	number of hexagons in a visibility map
N_p	number of particles in a particle filter
N_{sig}	number of sigma points
N_{TX}	number of signal components or transmitters
$N_{\text{TX},k}$	current number of signal components or transmitters at time instant k
$N^{\text{a}}_{\text{TX},k}$	overall number of signal components or transmitters at time instant k
$\boldsymbol{\tau}_k$	ToA estimates at time instant k
$\boldsymbol{\theta}_k$	AoA estimates at time instant k
$\boldsymbol{z}_{\text{R},k}$	radio measurement vector at time instant k
$\boldsymbol{z}_{\text{V},k}$	visibility measurement vector at time instant k
$\boldsymbol{u}_k$	control input at time instant k
$\boldsymbol{x}^{<j>}_{\text{TX},k}$	state of the j^{th} transmitter at time instant k
$\boldsymbol{p}^{<j>}_{\text{TX},k}$	location of the j^{th} transmitter at time instant k
$\tau^{<j>}_{0,k}$	propagation delay or clock offset of the j^{th} transmitter at time instant k
$\boldsymbol{x}_{\text{u},k}$	user state at time instant k
$\boldsymbol{x}_k$	state vector at time instant k in Channel-SLAM
$\boldsymbol{x}^{<i,j>}_{\text{TX},k}$	state vector of the j^{th} transmitter regarding the i^{th} user particle at time instant k
$\boldsymbol{x}^{<i>}_{\text{u},k}$	i^{th} user or user-map particle at time instant k
$w^{<i>}_k$	weight of $\boldsymbol{x}^{<i>}_{\text{u},k}$ at time instant k
$\boldsymbol{x}^{<i,j,\ell>}_{\text{TX},k}$	ℓ^{th} transmitter particle or mean of the ℓ^{th} Gaussian component regarding the i^{th} user particle and the j^{th} transmitter at time instant k
$w^{<i,j,\ell>}_k$	weight of $\boldsymbol{x}^{<i,j,\ell>}_{\text{TX},k}$ at time instant k
$\boldsymbol{P}^{<i,j,\ell>}_k$	covariance matrix of the ℓ^{th} Gaussian component regarding the i^{th} user particle and the j^{th} transmitter at time instant k

$V_h^{<j>}$	probability that the j^{th} transmitter is visible from the h^{th} hexagon
$\boldsymbol{V}_{h,k}^{<j>}$	random variable regarding $V_h^{<j>}$ at time instant k
$\boldsymbol{M}_k$	set of random variables $\boldsymbol{V}_{h,k}^{<j>}$ for all transmitters and hexagons at time instant k
$p_{h,k}^{<j>}$	parameter of a Beta distribution regarding the j^{th} transmitter was visible from the h^{th} hexagon
$q_{h,k}^{<j>}$	parameter of a Beta distribution regarding the j^{th} transmitter was not visible from the h^{th} hexagon
$C_{h,k}^{<j>}$	number of times the j^{th} transmitter has been visible from the h^{th} hexagon
$\bar{C}_{h,k}^{<j>}$	number of times the j^{th} transmitter has not been visible from the h^{th} hexagon
$\mathbb{E}_{h,k}^{<j>}$	expectation value of the Beta distribution regarding the j^{th} transmitter being visible in the h^{th} hexagon at time instant k
$\bar{\mathbb{E}}_{h,k}^{<j>}$	expectation value of the Beta distribution regarding the j^{th} transmitter not being visible in the h^{th} hexagon at time instant k
a_k	association variable at time instant k
$\mathcal{A}_k$	set of association decisions at time instant k
$\mathrm{p}_{e,\mathrm{u}}^{<j>}$	belief in the existence of the j^{th} transmitter
$\mathrm{d}_{\mathrm{rel,u}}^{<j>}$	reliability distance of the j^{th} transmitter
H_s	s^{th} RANSAC hypothesis
$\mathcal{C}_s$	set of correspondences regarding the s^{th} RANSAC hypothesis
$N_{\mathcal{C},s}$	size of the set of correspondences regarding the s^{th} RANSAC hypothesis
$\mathcal{H}$	set of RANSAC hypotheses

List of Figures

Acronyms

3GPP	3rd Generation Partnership Project
5G	fifth generation
AIC	Akaike information criterion
AoA	angle of arrival
AWGN	additive white Gaussian noise
BIC	Bayesian information criterion
CDF	cumulative distribution function
CIR	channel impulse response
DAS	data association sampling
DMC	dense multipath component
EKF	extended Kalman filter
EM	Expectation-Maximization
ESPRIT	Estimation of Signal Parameters via Rotational Invariance Techniques
GDoP	geometrical dilution of precision
GNSS	global navigation satellite system
GPS	Global Positioning System
GSF	Gaussian sum filter
GSPF	Gaussian sum particle filter
IMU	inertial measurement unit
KEST	Kalman Enhanced Super Resolution Tracking
LoS	line-of-sight
LTE	Long Term Evolution
MAE	mean absolute error
MAP	maximum a posteriori
MC	Monte Carlo
MHT	multiple hypothesis tracking
ML	maximum likelihood
MMSE	minimum mean square error

MPC	multipath component
MSE	mean square error
MUSIC	Multiple Signal Classification
NLoS	non-line-of-sight
PDF	probability density function
RANSAC	random sample consensus
RFID	radio-frequency identification
RSS	received signal strength
SAGE	Space-Alternating Generalized Expectation-Maximization
SLAM	simultaneous localization and mapping
SNR	signal-to-noise ratio
SoO	signal of opportunity
TDoA	time difference of arrival
ToA	time of arrival
UKF	unscented Kalman filter
UWB	ultra-wideband
WLAN	wireless local area network

Bibliography

[3rd13] 3rd Generation Partnership Project - Technical Specification Group Radio Access Network. *TR 36.808, Evolved Universal Terrestrial Radio Access (E-UTRA); Carrier Aggregation; Base Station (BS) radio transmission and reception (Release 10)*. Technical Report. V10.1.0. July 2013.

[AAR99] G.E. Andrews, R. Askey, and R. Roy. *Special Functions*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1999.

[AH09] I. Arasaratnam and Simon Haykin. "Cubature Kalman Filters". In: *IEEE Trans. Autom. Control* 54.6 (June 2009), pp. 1254–1269.

[AM12] B.D.O. Anderson and J.B. Moore. *Optimal Filtering*. Dover Books on Electrical Engineering. Dover Publications, 2012.

[AR12] M. Angermann and P. Robertson. "FootSLAM: Pedestrian Simultaneous Localization and Mapping Without Exteroceptive Sensors–Hitchhiking on Human Perception and Cognition". In: *Proceedings of the IEEE* 100.Special Centennial Issue (May 2012), pp. 1840–1848.

[Aru+02] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. "A Tutorial on Particle Filters for Online Nonlinear/non-Gaussian Bayesian Tracking". In: *IEEE Trans. Signal Process.* 50.2 (Feb. 2002), pp. 174–188.

[AT10] Gianni Amisano and Oreste Tristani. "Euro Area Inflation Persistence in an Estimated Nonlinear DSGE Model". In: *Journal of Economic Dynamics and Control* 34.10 (2010), pp. 1837–1858.

[Bav09] François Bavaud. "Information Theory, Relative Entropy and Statistics". In: *Formal Theories of Information: From Shannon to Semantic Information Theory and General Concepts of Information*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 54–78.

[BBS10] S. A. Berrabah, Y. Baudoin, and H. Sahli. "SLAM for Robotic Assistance to Fire-Fighting Services". In: *8th World Congress on Intelligent Control and Automation*. July 2010, pp. 362–367.

[BD06] Tim Bailey and H. Durrant-Whyte. “Simultaneous Localization and Mapping (SLAM): Part II”. In: *IEEE Robot. Autom. Mag.* 13.3 (Sept. 2006), pp. 108–117.

[Bd08] M. Bouet and A. L. dos Santos. “RFID Tags: Positioning Principles and Localization Techniques”. In: *2008 1st IFIP Wireless Days*. Nov. 2008, pp. 1–5.

[BK12] C. Beder and M. Klepal. “Fingerprinting Based Localisation Revisited: A Rigorous Approach for Comparing RSSI Measurements Coping with Missed Access Points and Differing Antenna Attenuations”. In: *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Nov. 2012, pp. 1–7.

[BLK04] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. Wiley, 2004.

[BOB07] Colin P.D. Birch, Sander P. Oom, and Jonathan A. Beecham. “Rectangular and Hexagonal Grids used for Observation, Experiment and Simulation in Ecology”. In: *Ecological Modelling* 206.3 (2007), pp. 347–359.

[BOO08] Francisco Bonin-Font, Alberto Ortiz, and Gabriel Oliver. “Visual Navigation for Mobile Robots: A Survey”. In: *Journal of Intelligent and Robotic Systems* 53.3 (2008), pp. 263–296.

[CA94] J. Chaffee and J. Abel. “GDOP and the Cramer-Rao Bound”. In: *Proceedings of IEEE/ION PLANS 1994*. Apr. 1994, pp. 663–668.

[Cad+16] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J.J. Leonard. “Past, Present, and Future of Simultaneous Localization And Mapping: Towards the Robust-Perception Age”. In: *IEEE Trans. Robot.* 32.6 (2016), pp. 1309–1332.

[Cal18] David Calvo Luz. “Design and Development of an Algorithm to Solve the Problem of Sharing Maps in Multipath Assisted Positioning”. MA thesis. Escuela Técnica Superior de Ingenieros de Telecomunicación of the Universidad Politécnica de Madrid, 2018.

[Che+04] Wen-Shiang Chen, Bhavik R. Bakshi, Prem K. Goel, and Sridhar Ungarala. “Bayesian Estimation via Sequential Monte Carlo Sampling: Unconstrained Nonlinear Dynamic Systems”. In: *Industrial & Engineering Chemistry Research* 43.14 (2004), pp. 4012–4025.

[CMK03] Ondřej Chum, Jiří Matas, and Josef Kittler. “Locally Optimized RANSAC”. In: *Pattern Recognition*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 236–243.

[CN10] G. Carneiro and J. C. Nascimento. “The Fusion of Deep Learning Architectures and Particle Filtering Applied to Lip Tracking”. In: *20th International Conference on Pattern Recognition*. Aug. 2010, pp. 2065–2068.

[CS13] Eleni N. Chatzi and Andrew W. Smyth. “Particle Filter Scheme with Mutation for the Estimation of Time-Invariant Parameters in Structural Health Monitoring Applications”. In: *Structural Control and Health Monitoring* 20.7 (2013), pp. 1081–1095.

[CT06] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications and Signal Processing. New York, NY, USA: Wiley-Interscience, 2006.

[DB06] H. Durrant-Whyte and Tim Bailey. “Simultaneous Localization and Mapping: Part I”. In: *IEEE Robot. Autom. Mag.* 13.2 (June 2006), pp. 99–110.

[DCC03] Nakju Doh, H. Choset, and Wan Kyun Chung. “Accurate Relative Localization Using Odometry”. In: *IEEE International Conference on Robotics and Automation*. Vol. 2. Sept. 2003, 1606–1612 vol.2.

[del+18] J. A. del Peral-Rosado, R. Raulefs, J. A. López-Salcedo, and G. Seco-Granados. “Survey of Cellular Mobile Radio Localization Methods: From 1G to 5G”. In: *IEEE Commun. Surveys Tuts.* 20.2 (2018), pp. 1124–1148.

[DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. “Maximum Likelihood from Incomplete Data via the EM Algorithm”. In: *Journal of the Royal Statistical Society. Series B (Methodological)* 39.1 (1977), pp. 1–38.

[DSR12] A. Dammann, S. Sand, and R. Raulefs. “Signals of Opportunity in Mobile Radio Positioning”. In: *Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*. Aug. 2012, pp. 549–553.

[Far12] Ramsey Faragher. “Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation [Lecture Notes]”. In: *IEEE Signal Process. Mag.* 29 (Sept. 2012), pp. 128–132.

[FB81] Martin A. Fischler and Robert C. Bolles. “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography”. In: *Commun. ACM* 24.6 (June 1981), pp. 381–395.

[Fle+96] B.H. Fleury, D. Dahlhaus, R. Heddergott, and M. Tschudin. “Wideband Angle of Arrival Estimation Using the SAGE Algorithm”. In: *IEEE 4th International Symposium on Spread Spectrum Techniques and Applications Proceedings*. Vol. 1. Sept. 1996, pp. 79–85.

[Fle+99] B. Fleury, M. Tschudin, R. Heddergott, D. Dahlhaus, and K. Pedersen. “Channel Parameter Estimation in Mobile Radio Environments Using the SAGE Algorithm”. In: *IEEE J. Sel. Areas Commun.* 17.3 (Mar. 1999), pp. 434–450.

[FR07] Jesús Fernández-Villaverde and Juan F. Rubio-Ramírez. "Estimating Macroeconomic Models: A Likelihood Approach". In: *The Review of Economic Studies* 74.4 (2007), pp. 1059–1087.

[FW88] M. Feder and E. Weinstein. "Parameter Estimation of Superimposed Signals Using the EM Algorithm". In: *IEEE Trans. Acoust., Speech, Signal Process.* 36.4 (Apr. 1988), pp. 477–489.

[Gen+14] Christian Gentner, Robert Pöhlmann, Thomas Jost, and Armin Dammann. "Multipath Assisted Positioning Using a Single Antenna with Angle of Arrival Estimations". In: *Proceedings of the 27th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2014)*. Tampa, FL, USA, Sept. 2014.

[Gen+15] Christian Gentner, Robert Pöhlmann, Markus Ulmschneider, Thomas Jost, and Armin Dammann. "Multipath Assisted Positioning for Pedestrians". In: *Proceedings of the 28th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2015)*. Tampa, FL, USA, Sept. 2015.

[Gen+16a] Christian Gentner, Thomas Jost, Wei Wang, Siwei Zhang, Armin Dammann, and Uwe-Carsten Fiebig. "Multipath Assisted Positioning with Simultaneous Localization and Mapping". In: *IEEE Trans. Wireless Commun.* 15.9 (Sept. 2016), pp. 6104–6117.

[Gen+16b] Christian Gentner, Boxiao Ma, Robert Pöhlmann, Thomas Jost, and Armin Dammann. "Simultaneous Localization and Mapping in Multipath Environments: Mapping and Reusing of Virtual Transmitters". In: *Proceedings of the 29th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2016)*. 2016.

[Gen+16c] Christian Gentner, Boxiao Ma, Markus Ulmschneider, Thomas Jost, and Armin Dammann. "Simultaneous Localization and Mapping in Multipath Environments". In: *Proceedings of IEEE/ION PLANS 2016*. Apr. 2016.

[Gen+17a] Christian Gentner, Robert Pöhlmann, Markus Ulmschneider, Thomas Jost, and Armin Dammann. "Simultaneous Localization and Mapping Using Terrestrial Multipath Signals, GNSS and Inertial Sensors". In: *Proceedings of the 30th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2017)*. 2017.

[Gen+17b] Christian Gentner, Robert Pöhlmann, Markus Ulmschneider, Thomas Jost, and Siwei Zhang. "Positioning Using Terrestrial Multipath Signals and Inertial Sensors". In: *Mobile Information Systems* (2017).

[Gen18] Christian Gentner. "Channel-SLAM: Multipath Assisted Positioning". PhD thesis. Ulm University, July 2018.

[GJD13a] Christian Gentner, Thomas Jost, and Armin Dammann. “Accurate Indoor Positioning Using Multipath Components”. In: *Proceedings of the 26th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2013)*. Nashville, TN, USA, Sept. 2013.

[GJD13b] Christian Gentner, Thomas Jost, and Armin Dammann. “Indoor Positioning Using Time Difference of Arrival between Multipath Components”. In: *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Montbeliard, France, Oct. 2013.

[Gro13] Paul D. Groves. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*. 2nd Edition. Artech House GNSS library. Artech House, 2013, XIX, 776 S.

[GSS93] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. “Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation”. In: *IEE Proceedings F - Radar and Signal Processing* 140.2 (Apr. 1993), pp. 107–113.

[GU17] Christian Gentner and Markus Ulmschneider. “Simultaneous Localization and Mapping for Pedestrians Using Low-Cost Ultra-Wideband System and Gyroscope”. In: *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. 2017.

[Gus+02] F. Gustafsson, F. Gunnarsson, Niclas Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund. “Particle Filters for Positioning, Navigation, and Tracking”. In: *IEEE Trans. Signal Process.* 50.2 (Feb. 2002), pp. 425–437.

[Gut+14] Felipe Guth, Luan Silveira, Silvia Botelho, Paulo Drews-Jr, and Pedro Ballester. “Underwater SLAM: Challenges, State of the Art, Algorithms and a New Biologically-Inspired Approach”. In: *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*. Aug. 2014, pp. 981–986.

[GZJ18] Christian Gentner, Siwei Zhang, and Thomas Jost. “Log-PF: Particle Filtering in Logarithm Domain”. In: *Journal of Electrical and Computer Engineering* (2018).

[HC16] S. He and S. G. Chan. “Wi-Fi Fingerprint-Based Indoor Positioning: Recent Advances and Comparisons”. In: *IEEE Commun. Surveys Tuts.* 18.1 (2016), pp. 466–490.

[HD16] Shoudong Huang and Gamini Dissanayake. “A Critique of Current Developments in Simultaneous Localization and Mapping”. In: *International Journal of Advanced Robotic Systems* 13.5 (Sept. 2016).

[HNM13] A. Hast, J. Nysjo, and A. Marchetti. “Optimal RANSAC - Towards a Repeatable Algorithm for Finding the Optimal Set”. In: *Journal of WSCG* 1 (2013), pp. 21–30.

[HSE17] Oliver Heirich, Benjamin Siebler, and Hedberg Erik. “Study of Train-side Passive Magnetic Measurements with Applications to Train Localization”. In: *Journal of Sensors* (June 2017).

[IEE16] IEEE Computer Society. “IEEE Standard for Information technology—Telecommunications and information exchange between systems; Local and metropolitan area networks—Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”. In: *IEEE Std 802.11-2016 (Revision of IEEE Std. 802.11-2012)* (Dec. 2016).

[Jos+12] T. Jost, Wei Wang, U. Fiebig, and F. Perez-Fontan. “Detection and Tracking of Mobile Propagation Channel Paths”. In: *IEEE Trans. Antennas Propag.* 60.10 (Oct. 2012), pp. 4875–4883.

[Jos14] Thomas Jost. “Satellite-to-Indoor Wave Propagation for Positioning Applications”. PhD thesis. University of Vigo, 2014.

[Jou+16] Marine Jouin, Rafael Gouriveau, Daniel Hissel, Marie-Cécile Péra, and Noureddine Zerhouni. “Particle Filter-Based Prognostics: Review, Discussion and Perspectives”. In: *Mechanical Systems and Signal Processing* 72-73 (2016), pp. 2–31.

[JS01] S. Rao Jammalamadaka and Ashis Sengupta. *Topics in Circular Statistics*. Vol. 5. Series on Multivariate Analysis. World Scientific, Apr. 2001.

[JU04] Simon J. Julier and Jeffrey K. Uhlmann. “Unscented Filtering and Nonlinear Estimation”. In: *Proceedings of the IEEE*. 2004, pp. 401–422.

[JU97] S.J. Julier and J.K. Uhlmann. “A New Extension of the Kalman Filter to Nonlinear Systems”. In: *Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defense Sensing, Simulations and Controls*. 1997.

[JUD95] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte. “A New Approach for Filtering Nonlinear Systems”. In: *Proceedings of the American Control Conference*. Vol. 3. June 1995, pp. 1628–1632.

[Jul02] S. J. Julier. “The Scaled Unscented Transformation”. In: *Proceedings of the 2002 American Control Conference*. Vol. 6. May 2002, 4555–4559 vol.6.

[Kal60] R. E. Kalman. “A New Approach to Linear Filtering and Prediction Problems”. In: *Journal of Basic Engineering* 82.1 (Mar. 1960), pp. 35–45.

[KAT13] Yubin Kuang, K. Astrom, and F. Tufvesson. “Single Antenna Anchor-Free UWB Positioning based on Multipath Propagation”. In: *IEEE International Conference on Communications (ICC)*. June 2013, pp. 5814–5818.

[Kay98] S.M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory*. Fundamentals of Statistical Signal Processing. Prentice-Hall PTR, 1998.

[Kel03] Alonzo Kelly. "Precision Dilution in Triangulation based Mobile Robot Position Estimation". In: *In Intelligent Autonomous Systems*. 2003.

[KG20] Rostislav Karásek and Christian Gentner. "Stochastic Data Association for Multipath Assisted Positioning Using a Single Transmitter". In: *IEEE Access* 8 (Mar. 2020), pp. 46735–46752.

[KH05] Elliott Kaplan and Christopher Hegarty. *Understanding GPS - Principles and Applications*. Second Edition. Artech House, Dec. 2005.

[Kim+18] H. Kim, H. Wymeersch, N. Garcia, G. Seco-Granados, and S. Kim. "5G mmWave Vehicular Tracking". In: *52nd Asilomar Conference on Signals, Systems, and Computers*. Oct. 2018, pp. 541–547.

[Kle04] L. A. Klein. *Sensor and Data Fusion: A Tool for Information Assessment and Decision Making*. Press Monographs. Society of Photo Optical, 2004.

[KLW94] Augustine Kong, Jun S. Liu, and Wing Hung Wong. "Sequential Imputations and Bayesian Missing Data Problems". In: *Journal of the American Statistical Association* 89.425 (Mar. 1994), pp. 278–288.

[Kot14] Krishna Kumar Kottakki. "Gaussian Sum Unscented Gaussian Sum Filter (An Improvement in GSUKF)". In: *Third International Conference on Advances in Control and Optimization of Dynamical Systems*. Mar. 2014.

[KS11] Jonathan Kelly and Gaurav S. Sukhatme. "Visual-Inertial Sensor Fusion: Localization, Mapping and Sensor-to-Sensor Self-Calibration". In: *The International Journal of Robotics Research* 30.1 (2011), pp. 56–79.

[KSC07] C. Kim, R. Sakthivel, and W. K. Chung. "Unscented FastSLAM: A Robust Algorithm for the Simultaneous Localization and Mapping Problem". In: *Proceedings of the IEEE International Conference on Robotics and Automation*. Apr. 2007, pp. 2439–2445.

[KSC08] C. Kim, R. Sakthivel, and W. K. Chung. "Unscented FastSLAM: A Robust and Efficient Solution to the SLAM Problem". In: *IEEE Trans. Robot.* 24.4 (Aug. 2008), pp. 808–820.

[KV96] H. Krim and M. Viberg. "Two Decades of Array Signal Processing Research: The Parametric Approach". In: *IEEE Signal Processing Magazine* 13.4 (July 1996), pp. 67–94.

[Lei+14] E. Leitinger, M. Fröhle, P. Meissner, and K. Witrisal. "Multipath-Assisted Maximum-Likelihood Indoor Positioning Using UWB Signals". In: *IEEE International Conference on Communications Workshops (ICC Workshops 2014)*. June 2014, pp. 170–175.

[Lei+15] E. Leitinger, P. Meissner, C. Rüdisser, G. Dumphart, and K. Witrisal. "Evaluation of Position-Related Information in Multipath Components for Indoor Positioning". In: *IEEE J. Sel. Areas Commun.* 33.11 (Nov. 2015), pp. 2313–2328.

[Lei+16] E. Leitinger, F. Meyer, P. Meissner, K. Witrisal, and F. Hlawatsch. "Belief Propagation Based Joint Probabilistic Data Association for Multipath-Assisted Indoor Navigation and Tracking". In: *2016 International Conference on Localization and GNSS (ICL-GNSS)*. June 2016, pp. 1–6.

[Lei+17] E. Leitinger, F. Meyer, F. Tufvesson, and K. Witrisal. "Factor Graph Based Simultaneous Localization and Mapping Using Multipath Channel Information". In: *IEEE International Conference on Communications Workshops (ICC Workshops 2017)*. May 2017, pp. 652–658.

[Lei+19] E. Leitinger, F. Meyer, F. Hlawatsch, K. Witrisal, F. Tufvesson, and M. Z. Win. "A Belief Propagation Algorithm for Multipath-Based SLAM". In: *IEEE Trans. Wireless Commun.* 18.12 (2019), pp. 5613–5629.

[Leo+13] P.H. Leong, S. Arulampalam, T.A. Lamahewa, and T.D. Abhayapala. "A Gaussian-Sum Based Cubature Kalman Filter for Bearings-Only Tracking". In: *IEEE Trans. Aerosp. Electron. Syst.* 49.2 (Apr. 2013), pp. 1161–1176.

[LK06] Michael Lentmaier and Bernhard Krach. "Maximum Likelihood Multipath Estimation in Comparison with Conventional Delay Lock Loops". In: *Proceedings of the 19th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2006)*. Sept. 2006.

[Low+16] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford. "Visual Place Recognition: A Survey". In: *IEEE Trans. Robot.* 32.1 (Feb. 2016), pp. 1–19.

[MC05] Ruben Martinez-Cantin and José A. Castellanos. "Unscented SLAM for Large-Scale Outdoor Environments". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems* (2005), pp. 3427–3432.

[MCS02] Mun Wai Lee, I. Cohen, and Soon Ki Jung. "Particle Filter with Analytical Inference for Human Body Tracking". In: *Workshop on Motion and Video Computing, 2002. Proceedings.* Dec. 2002, pp. 159–165.

[ME11] P. Misra and P. Enge. *Global Positioning System: Signals, Measurements, and Performance*. Ganga-Jamuna Press, 2011.

[Mei+13] Paul Meissner, Erik Leitinger, Markus Fröhle, and Klaus Witrisal. "Accurate and Robust Indoor Localization Systems Using Ultra-Wideband Signals". In: *European Conference on Navigation (ENC)*. 2013.

[Men+09] E. Menegatti, A. Zanella, S. Zilli, F. Zorzi, and E. Pagello. "Range-only SLAM with a Mobile Robot and a Wireless Sensor Networks". In: *2009 IEEE International Conference on Robotics and Automation*. May 2009, pp. 8–14.

[Men+19a] R. Mendrzik, F. Meyer, G. Bauch, and M. Win. "Localization, Mapping, and Synchronization in 5G Millimeter Wave Massive MIMO Systems". In: *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. 2019, pp. 1–5.

[Men+19b] R. Mendrzik, H. Wymeersch, G. Bauch, and Z. Abu-Shaban. "Harnessing NLOS Components for Position and Orientation Estimation in 5G Millimeter Wave MIMO". In: *IEEE Trans. Wireless Commun.* 18.1 (Jan. 2019), pp. 93–107.

[MGW10] P. Meissner, T. Gigl, and K. Witrisal. "UWB Sequential Monte Carlo Positioning Using Virtual Anchors". In: *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Sept. 2010, pp. 1–10.

[Mon+02] Michael Montemerlo, Sebastian Thrun, Daphne Koller, and Ben Wegbreit. "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem". In: *In Proceedings of the AAAI National Conference on Artificial Intelligence*. AAAI, 2002, pp. 593–598.

[Mon+03] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. "FastSLAM 2.0: An Improved Particle Filtering Algorithm for Simultaneous Localization and Mapping that Provably Converges". In: *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI. Acapulco, Mexico, 2003.

[Mül+17] P. Müller, S. Ali-Löytty, J. Lekkala, and R. Piché. "Indoor Localisation using Aroma Fingerprints: A First Sniff". In: *14th Workshop on Positioning, Navigation and Communications (WPNC)*. Oct. 2017, pp. 1–5.

[NCR19] David Nakath, Joachim Clemens, and Carsten Rachuy. "Active Asteroid-SLAM". In: *Journal of Intelligent & Robotic Systems* (Nov. 2019).

[NK17] H. Naseri and V. Koivunen. "Cooperative Simultaneous Localization and Mapping by Exploiting Multipath Propagation". In: *IEEE Trans. Signal Process.* 65.1 (Jan. 2017), pp. 200–211.

[OSD15] A. O'Connor, P. Setlur, and N. Devroye. "Single-Sensor RF Emitter Localization Based on Multipath Exploitation". In: *IEEE Trans. Aerosp. Electron. Syst.* 51.3 (July 2015), pp. 1635–1651.

[PRK85] A. Paulraj, R. Roy, and T. Kailath. “Estimation Of Signal Parameters Via Rotational Invariance Techniques- Esprit”. In: *Nineteeth Asilomar Conference on Circuits, Systems and Computers*. Nov. 1985, pp. 83–89.

[PS96] B.W. Parkinson and J.J. Spilker. *Global Positioning System: Theory and Applications*. Bd. 1. American Institute of Aeronautics & Astronautics, 1996.

[QK02] Y. Qi and H. Kobayashi. “Cramér-Rao Lower Bound for Geolocation in Non-Line-of-Sight Environment”. In: *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 3. May 2002, pp. III-2473-III–2476.

[RAG04] Branko Ristic, Sanjeev Arulampalam, and Neil Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2004.

[RFP08] Rahul Raguram, Jan-Michael Frahm, and Marc Pollefeys. “A Comparative Analysis of RANSAC Techniques Leading to Adaptive Real-Time Random Sample Consensus”. In: *Computer Vision – ECCV 2008*. 2008, pp. 500–513.

[Ric05] Andreas Richter. “Estimation of Radio Channel Parameters: Models and Algorithms”. PhD thesis. Technische Universität Ilmenau, 2005.

[RR97] I. A. Rezek and S. J. Roberts. “Parametric Model Order Estimation: A Brief Review”. In: *IEE Colloquium on Model Based Digital Signal Processing Techniques in the Analysis of Biomedical Signals (Digest No. 1997/009)*. Apr. 1997, pp. 3/1–3/6.

[RRL99] J. Razavilar, F. Rashid-Farrokhi, and K. J. R. Liu. “Software Radio Architecture with Smart Antennas: A Tutorial on Algorithms and Complexity”. In: *IEEE J. Sel. Areas Commun.* 17.4 (Apr. 1999), pp. 662–676.

[Sär+16] Simo Särkkä, Jouni Hartikainen, Lennart Svensson, and Fredrik Sandblom. “On the Relation Between Gaussian Process Quadratures and Sigma-Point Methods”. In: *Journal of Advances in Information Fusion* 11.1 (June 2016), pp. 31–46.

[Sär13] Simo Särkkä. *Bayesian Filtering and Smoothing*. New York, NY, USA: Cambridge University Press, 2013.

[Sch03] Alexander Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*. Springer, 2003.

[Sch86] R. Schmidt. “Multiple Emitter Location and Signal Parameter Estimation”. In: *IEEE Trans. Antennas Propag.* 34.3 (Mar. 1986), pp. 276–280.

[SD13] Pawan Setlur and Natasha Devroye. “Multipath Exploited Bayesian and Cramer-Rao Bounds for Single-Sensor Target Localization”. In: *EURASIP Journal on Advances in Signal Processing* 2013 (2013).

[Set+12] P. Setlur, G.E. Smith, F. Ahmad, and M.G. Amin. "Target Localization with a Single Sensor via Multipath Exploitation". In: *IEEE Trans. Aerosp. Electron. Syst.* 48.3 (July 2012), pp. 1996–2014.

[SF11] D. Scaramuzza and F. Fraundorfer. "Visual Odometry [Tutorial]". In: *IEEE Robot. Autom. Mag.* 18.4 (Dec. 2011), pp. 80–92.

[SG11] Eric Schenk and Claude Guittard. "Towards a Characterization of Crowdsourcing Practices". In: *Journal of Innovation Economics & Management* 7.1 (2011), pp. 93–107.

[SH74] Ivan E. Sutherland and Gary W. Hodgman. "Reentrant Polygon Clipping". In: *Commun. ACM* 17.1 (Jan. 1974), pp. 32–42.

[Sha+18] A. Shahmansoori, G. E. Garcia, G. Destino, G. Seco-Granados, and H. Wymeersch. "Position and Orientation Estimation Through Millimeter-Wave MIMO in 5G Systems". In: *IEEE Trans. Wireless Commun.* 17.3 (2018), pp. 1822–1835.

[She+06] Hailin Shen, Glyn Nelson, Stephnie Kennedy, David Nelson, James Johnson, David Spiller, Michael R.H. White, and Douglas B. Kell. "Automatic Tracking of Biological Cells and Compartments Using Particle Filters and Active Contours". In: *Chemometrics and Intelligent Laboratory Systems* 82.1 (2006). Selected Papers from the International Conference on Chemometrics and Bioinformatics in Asia, pp. 276–282.

[Sil86] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. London: Chapman & Hall, 1986.

[Sim06] Dan Simon. *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. Wiley-Interscience, 2006.

[SM16] Guowei Shi and Ying Ming. "Survey of Indoor Positioning Systems Based on Ultra-Wideband (UWB) Technology". In: *Wireless Communications, Networking and Applications*. 2016, pp. 1269–1278.

[SR96] M. I. Silventoinen and T. Rantalainen. "Mobile Station Emergency Locating in GSM". In: *IEEE International Conference on Personal Wireless Communications Proceedings and Exhibition*. Feb. 1996, pp. 232–238.

[SS03] Jim Smith and Antonio Santos. "Second-Order Filter Distribution Approximations for Financial Time Series With Extreme Outliers". In: *Journal of Business and Economic Statistics* 24 (Feb. 2003).

[SS04] Petre Stoica and Y. Selen. "Model-Order Selection: A Review of Information Criterion Rules". In: *IEEE Signal Process. Mag.* 21.4 (July 2004), pp. 36–47.

[SSR10] W. Storms, J. Shockley, and J. Raquet. “Magnetic Field Navigation in an Indoor Environment”. In: *2010 Ubiquitous Positioning Indoor Navigation and Location Based Service*. Oct. 2010, pp. 1–10.

[SW09] Yuan Shen and M.Z. Win. “On the Use of Multipath Geometry for Wideband Cooperative Localization”. In: *IEEE Global Telecommunications Conference (GLOBECOM)*. Nov. 2009, pp. 1–6.

[Sze06] Richard Szeliski. “Image Alignment and Stitching: A Tutorial”. In: *Found. Trends. Comput. Graph. Vis.* 2.1 (Jan. 2006), pp. 1–104.

[TBF05] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. Intelligent Robotics and Autonomous Agents. MIT Press, 2005.

[Thr+04] S. Thrun, M. Montemerlo, D. Koller, B. Wegbreit, J. Nieto, and E. Nebot. “FastSLAM: An Efficient Solution to the Simultaneous Localization And Mapping Problem with Unknown Data Association”. In: *Journal of Machine Learning Research* 4.3 (2004), pp. 380–407.

[Thr+05] Sebastian Thrun, S. Thayer, William Whittaker, Christopher Baker, Wolfram Burgard, D. Ferguson, D Hähnel, M Montemerlo, A.C. Morris, Zachary Omohundro, and C. Reverte. “Autonomous Exploration and Mapping of Abandoned Mines”. In: *IEEE Robot. Autom. Mag.* (Jan. 2005).

[Tsc+99] M. Tschudin, C. Brunner, T. Kurpjuhn, M. Haardt, and J. A. Nossek. “Comparison between Unitary ESPRIT and SAGE for 3-D Channel Sounding”. In: *1999 IEEE 49th Vehicular Technology Conference*. Vol. 2. May 1999, 1324–1329 vol.2.

[Tur60] G. Turin. “An Introduction to Matched Filters”. In: *IRE Transactions on Information Theory* 6.3 (June 1960), pp. 311–329.

[TWW04] D. Titterton, J.L. Weston, and J. Weston. *Strapdown Inertial Navigation Technology*. Electromagnetics and Radar Series. Institution of Engineering and Technology, 2004.

[UG18] Markus Ulmschneider and Christian Gentner. “Improving Maps of Physical and Virtual Radio Transmitters”. In: *Proceedings of the 31st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2018)*. Sept. 2018.

[UG19] Markus Ulmschneider and Christian Gentner. “RANSAC for Exchanging Maps in Multipath Assisted Positioning”. In: *IEEE International Conference on Industrial Cyber Physical Systems (ICPS)*. 2019.

[UGD19] Markus Ulmschneider, Christian Gentner, and Armin Dammann. “Data Association among Physical and Virtual Radio Transmitters with Visibility Regions”. In: *IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*. Sept. 2019.

[UGD20] Markus Ulmschneider, Christian Gentner, and Armin Dammann. "Matching Maps of Physical and Virtual Radio Transmitters Using Visibility Regions". In: *Proceedings of IEEE/ION PLANS 2020*. May 2020.

[ULG18] Markus Ulmschneider, David Calvo Luz, and Christian Gentner. "Exchanging Transmitter Maps in Multipath Assisted Positioning". In: *Proceedings of IEEE/ION PLANS 2018*. 5, 2018.

[Ulm+15] Markus Ulmschneider, Christian Gentner, Simon Ache, and Andreas Roessler. "Multipath Assisted Positioning with Band-Limited Signals in an Urban Environment". In: *Proceedings of the 28th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2015)*. Sept. 2015, pp. 2329–2334.

[Ulm+17a] Markus Ulmschneider, Christian Gentner, Thomas Jost, and Armin Dammann. "Association of Transmitters in Multipath-Assisted Positioning". In: *IEEE Global Communications Conference (GLOBECOM)*. Dec. 2017.

[Ulm+17b] Markus Ulmschneider, Christian Gentner, Thomas Jost, and Armin Dammann. "Multiple Hypothesis Data Association for Multipath-Assisted Positioning". In: *14th Workshop on Positioning, Navigation and Communications (WPNC)*. Oct. 2017.

[Ulm+18] Markus Ulmschneider, Christian Gentner, Thomas Jost, and Armin Dammann. "Rao-Blackwellized Gaussian Sum Particle Filtering for Multipath Assisted Positioning". In: *Journal of Electrical and Computer Engineering* (2018).

[Ulm+20] Markus Ulmschneider, Siwei Zhang, Christian Gentner, and Armin Dammann. "Multipath Assisted Positioning With Transmitter Visibility Information". In: *IEEE Access* 8 (2020), pp. 155210–155223.

[Wan14] Wei Wang. "Channel Measurement and Modeling for Mobile Radio Based Positioning". PhD thesis. Friedrich-Alexander Universität Erlangen-Nürnberg, 2014.

[WH96] M. P. Wylie and J. Holtzman. "The Non-Line of Sight Problem in Mobile Location Estimation". In: *Proceedings of ICUPC - 5th IEEE International Conference on Universal Personal Communications*. Vol. 2. Oct. 1996, 827–831 vol.2.

[Wit+16] K. Witrisal, P. Meissner, E. Leitinger, Y. Shen, C. Gustafson, F. Tufvesson, K. Haneda, D. Dardari, A. F. Molisch, A. Conti, and M. Z. Win. "High-Accuracy Localization for Assisted Living - 5G Systems will turn Multipath Channels from Foe to Friend". In: *IEEE Signal Process. Mag.* 33.2 (Mar. 2016), pp. 59–70.

[WM00] E. A. Wan and R. Van Der Merwe. "The Unscented Kalman Filter for Nonlinear Estimation". In: *Proceedings of the IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium*. 2000, pp. 153–158.

[WM12] K. Witrisal and P. Meissner. “Performance Bounds for Multipath-Assisted Indoor Navigation and Tracking (MINT)”. In: *IEEE International Conference on Communications (ICC)*. June 2012.

[Wym+18] H. Wymeersch, N. Garcia, H. Kim, G. Seco-Granados, S. Kim, F. Wen, and M. Fröhle. “5G mm Wave Downlink Vehicular Positioning”. In: *IEEE Global Communications Conference (GLOBECOM)*. Dec. 2018, pp. 206–212.

[Yas+18] A. Yassin, Y. Nasser, A. Y. Al-Dubai, and M. Awad. “MOSAIC: Simultaneous Localization and Environment Mapping Using mmWave Without A-Priori Knowledge”. In: *IEEE Access* 6 (2018), pp. 68932–68947.

[YSG09] Kegen Yu, Ian Sharp, and Y. Jay Guo. *Ground-Based Wireless Positioning*. Wiley-IEEE Press, 2009.

[ZGL19] F. Zafari, A. Gkelias, and K. K. Leung. “A Survey of Indoor Localization Systems and Technologies”. In: *IEEE Commun. Surveys Tuts.* 21.3 (2019), pp. 2568–2599.